L'INFORMATIQUE

Premier contact

par Jean-Pierre Petit

L'INFORMATIQUE

Premier contact

Jean-Pierre Petit

CEDIC/FERNAND NATHAN

Ce volume porte le n° ISBN 2-7124-0608-7

Toute reproduction, même partielle, de cet ouvrage est interdite. Une copie ou reproduction par quelque procédé que ce soit, photographie, photocopie, microfilm, bande magnétique, disque ou autre, constitue une contrefaçon passible des peines prévues par la loi du 11 mars 1975 sur la protection des droits d'auteur.

© CEDIC 1983

CEDIC, 32, boulevard Saint-Germain, 75005 - PARIS

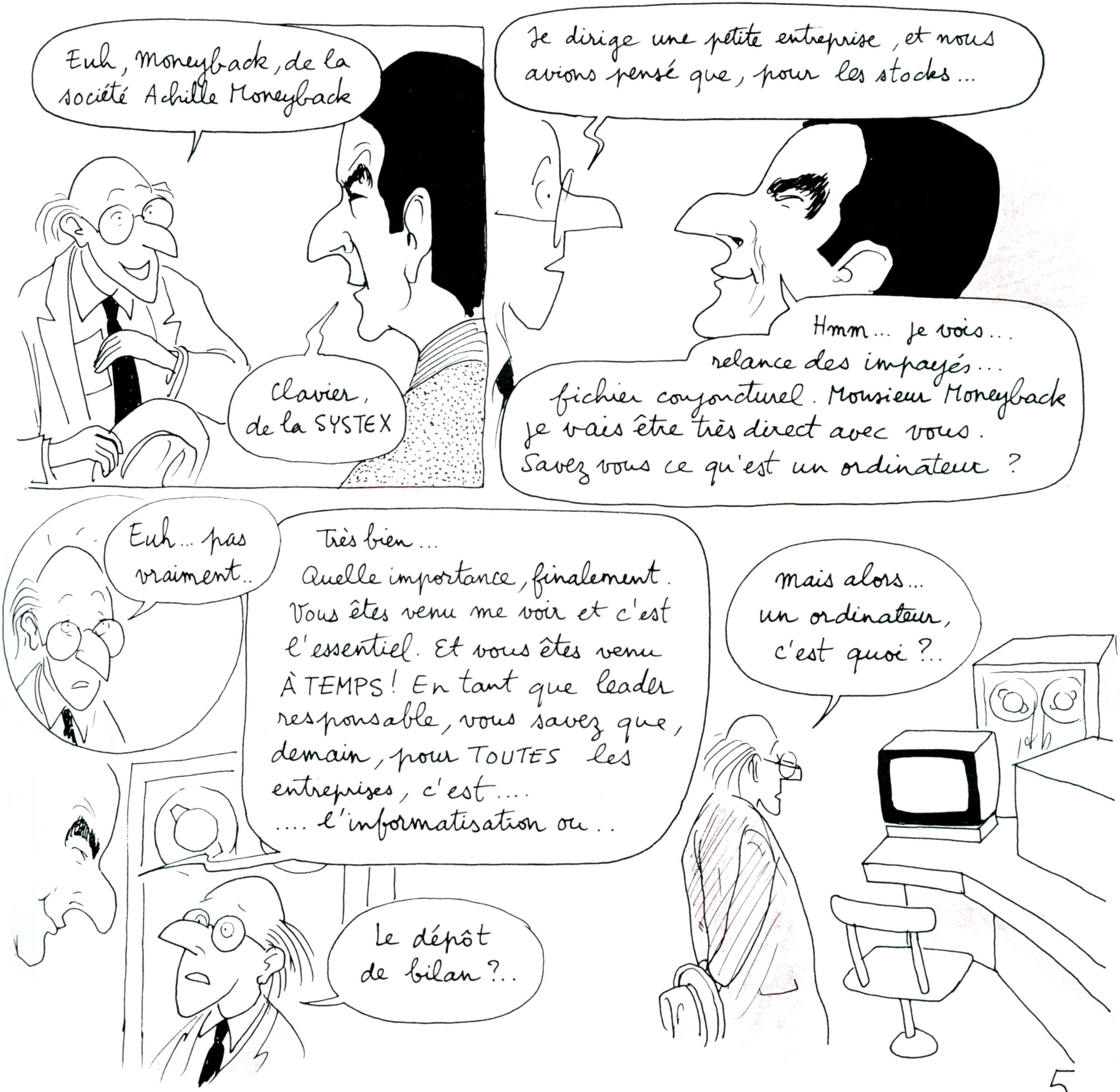
GLOSSAIRE INFORMATIQUE

3	INFORMATIQUE	Technique qui utilise l'ordinateur.
5	ORDINATEUR	Machine programmable de multiples façons.
6	LOGICIEL	Ensemble de programmes disponibles.
6	DISQUE LOURD	Mémoire externe à très forte capacité.
8	MICRO-ORDINATEUR	Ordinateur dont le "cerveau" est un microprocesseur.
8	MICROPROCESSEUR	Cerveau, ou plutôt cervelet d'un ordinateur.
9	UNITÉ CENTRALE	autre dénomination du "cervelet".
11	DISKETTE	mémoire externe sous forme d'un disque souple, bon marché.
11	ÉCRAN VIDÉO	Ecran TV pour afficher textes, chiffres, graphiques.
12	MEMOIRE	Dans l'ordinateur, peut stocker des nombres, des chaines de caractères, des instructions composant des programmes.
13	ADRESSE	Système de repérage des contenus.
14	REGISTRE	Element, case mémoire, contenant une DONNÉE : Nombre ou chaine.
14	CHAINE DE CARACTÈRE	Suite quelconque de signes.
14	DONNÉE	Information (Numérique ou Chaine) susceptible d'être introduite dans l'ordinateur avant ou pendant l'exécution d'un programme.
14	INSTRUCTION	opération élementaire codée, mémorisable.
14	LANGAGE	Ensemble d'instructions (qui sont les "mots" de ce langage).
14	SYNTAXE	règles d'utilisation des instructions.
15	MESSAGE D'ERREUR	Sanctionne une erreur (en général de syntaxe).
14	ÉCRITURE EN MÉMOIRE	Introduction de qqch dans la mémoire.
14	LECTURE	Interroger la mémoire, lire un registre, afficher son contenu.
17	ROM	Architecture de l'ordinateur, ineffaçable, où l'utilisateur n'a pas accès.
17	RAM	Zone mémoire où on peut ECRIRE (données ou programme).
19	PROGRAMME	Suite ordonnée d'instructions, présentées en LIGNES-PROGRAMMES.
22	LANCER UN PROGRAMME	Déclencher son exécution.
21	BASIC	Un langage particulier des ordinateurs.
27	LISTER UN PROGRAMME	Afficher ce programme, ligne par ligne, sur l'écran.

30	ENTRER DES DONNÉES	: Alimenter les registres de l'ordinateur.
34	SAUVER UN PROGRAMME	: Recopier un programme, à partir d'une mémoire centrale (volatile) sur une mémoire externe (non volatile), par exemple diskette.
34	MÉMOIRE VOLATILE	: Contenu détruit en mettant le système hors tension.
34	CHARGER UN PROGRAMME	: Recopier un programme de la mémoire externe vers la m. centrale.
36	BOUCLE	: Structure d'une tâche répétitive dans l'ordinateur.
38	INCRÉMENTATION	: augmentation du contenu d'un registre numérique.
38	INCRÉMENT	: La quantité dont on INCRÉMENTE le registre-compteur.
38	DÉCRÉMENTER, DÉCRÉMENT	: opération inverse.
39	INSTRUCTION DE BRANCHEMENT	: Envoit vers une autre ligne-programme.
39	BOUCLE SANS FIN	: Structure où rien n'a été prévu pour sortir de boucle.
39	BRANCHEMENT INCONDITIONNEL	: S'effectue à l'aide d'un GOTO
40	BRANCHEMENT CONDITIONNEL	: Assorti d'un Test : IF ..... THEN....
42	BOUCLE DE TEMPORISATION	: boucle "vide", qui maintient l'affichage sur écran.
51	OPTIONS	: Différentes sections de programme vers lesquelles on peut s'orienter.
52	MENU	: Structure de Programme permettant de multiples options.
53	SOUS-PROGRAMME	: Section de programme qui peut être, dans le programme principal, exécutée "toutes affaires cessantes", ceci étant suivi d'une reprise du cours normal.
57	TRAITEMENT DE TEXTE	: Manipulations de chaines de caractères.
59	SOUS-CHAINE	: Un fragment d'une chaine de caractères.
60	VARIABLE	: Registre dont le contenu peut varier en cours d'exécution.
61	VARIABLE INDICÉE	: Registre à contenu variable, repéré à l'aide d'un indice.
65	FICHIER	: Ensemble de registres repérés avec le (ou les) même indice.
67	RÉSERVATION DE PLACES MÉMOIRE	: Affectation d'un secteur de la mémoire pour y loger un ou des fichiers, à un ou plusieurs indices.
74	FLAG	: Registre où on note, d'une manière quelconque, si une opération a été effectuée ou non.

ma bichette
oui, mon ami, vous me l'avez déja dit cent fois
c'est l'informatisation ou le dépôt de bilan
mais comme personne dans cette maison n'a été capable de me dire ce qu'était l'informatique, je me contenterai de bien penser à vous
Au SICOB, s'il vous plait
allons, viens, Dagobert !
Voyons, le petit Larousse INFORMATIQUE : technique de traitement automatique de l'information
mon Dieu je sens que cela va être dur

VOUS ETES ICI
Courage...
PAR LÀ
S'il vous plait, où est-ce qu'on peut voir des ordinateurs ?
Ca dépend. Que voulez vous voir ? Des minis ? des micros ?
Euh... des moyens...
Voyons... troisième niveau... ..porte G ... stands huit à onze...
PIEGEAC 2000
ALOUETTE PLACART
DALLAS INSTRUMENT
le pigeon-type
un petit conseil en passant, monsieur ?


Euh, Moneyback, de la société Achille Moneyback
Clavier, de la SYSTEX
Je dirige une petite entreprise, et nous avions pensé que, pour les stocks...
Hmm... je vois... relance des impayés... fichier conjoncturel. Monsieur Moneyback je vais être très direct avec vous. Savez vous ce qu'est un ordinateur ?
Euh... pas vraiment..
Très bien... Quelle importance, finalement. Vous êtes venu me voir et c'est l'essentiel. Et vous êtes venu À TEMPS! En tant que leader responsable, vous savez que, demain, pour TOUTES les entreprises, c'est.... .... l'informatisation ou..
Le dépôt de bilan?..
mais alors... un ordinateur, c'est quoi?..

Je schématise. Vous voyez ce rond. Cela sera pour nous le système ENTRÉE-SORTiE que constitue l'ordinateur.
mais, l'entrée-sortie c'est QUOi?...
Entrée, traitement, sortie. Ce sont les mots-clefs. Vous n'avez pas besoin d'en savoir plus!...
ENTRÉE
SORTiE
TRAITEMENT
Ah!..
....notre nouveau PiÈGEAC 2000, avec son puissant logiciel polyvalent, entièrement conversationnel, avec saisie directe assistée et comptabilisation en temps réel, accès multiposte, disc mégaoctets en ligne, s'occupe du reste...
Mais, son prix?
Pour une configuration utilisable avec deux disques lourds pour commencer, en comptant la maintenance de logiciel, cela ferait...

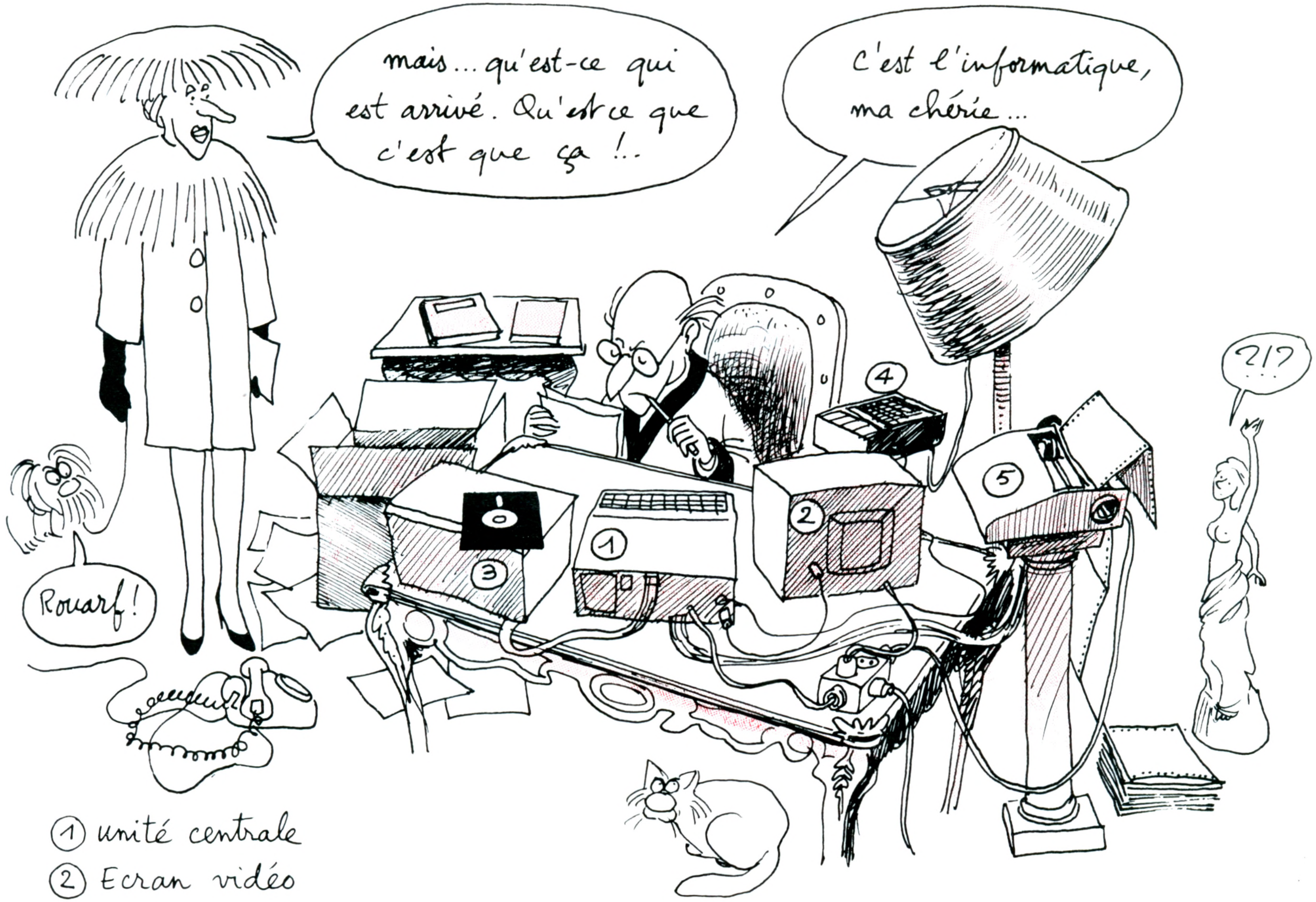
Zut, mes piles sont à plat !
Ne pourrait-on pas utiliser votre merveilleuse machine ... votre PIEGEAC ?..
mais... bien sûr
Zut ! la tuile...
Bon, c'est simple, je charge l'operating system par CC300/A ce qui libère mon WORK SPACE
Ah, zut, il est en BATCH ! J'ai oublié de faire TTX c'est à dire une demande d'interruption prioritaire
Vous comprenez ?
Euh... oui !

l'informatique...
je n'y comprendrai jamais rien
LE FORTRAN DANS LA JOIE
PRATIQUE DU 6502
LE PASCAL SANS PEINE
JE PARLE LE COBOL
Ouh là là....
Ben dis donc...
SBOUNG!
!?
B
Excusez moi, messieurs, mais qu'est-ce que cet appareil fait au SICOB?
C'est un microordinateur
Un ordinateur? on dirait une machine à sous!
Mais non, tout y est, l'UNITÉ CENTRALE, le MICROPROCESSEUR, les MÉMOIRES.
mais... je ne vois qu'un clavier et un écran de télévision

Où est... le reste ?
Tout est là
Tenez, voilà l'unité centrale
Ce sont ces petites choses
qu'on trouve dans les jouets ?
Pas seulement. Un ordinateur peut faire des tas de choses très variées, vous savez...
Je passe une DISKETTE
?
Ça par exemple c'est un programme qu'a fait un copain du CNRS. Ça calcule des champs magnétiques.
CHAMP BØ=1ØØØ GAUSS
DEUX BARRES A+/- 2CM
Ah bon, ça fait aussi du calcul scientifique !
Mais... pas seulement. Tenez, avec cet autre programme je transforme l'ordinateur en synthétiseur musical.
Mais... la musique est sur ce... disque ?
Oui, mais pas comme vous croyez. Ça n'est pas un 45 tours !

maintenant vous le faites .... parler !?!!
Je suis le synthétiseur vocal ...
Avec un micro ordinateur on peut faire du dessin industriel, contrôler un processus expérimental, ou une machine-outil.
On peut lui faire taper des lettres, gérer des stocks, des fichiers, lui faire faire des statistiques.
AH...
TILT...
Cette fois-ci je plonge !..
Hep, Tasci !

L'ORDINATEUR A LA MAISON



1. unité centrale
2. Ecran vidéo
3. lecteur de diskettes
4. magnétophone
5. Imprimante

Achille Moneyback soupira profondément.

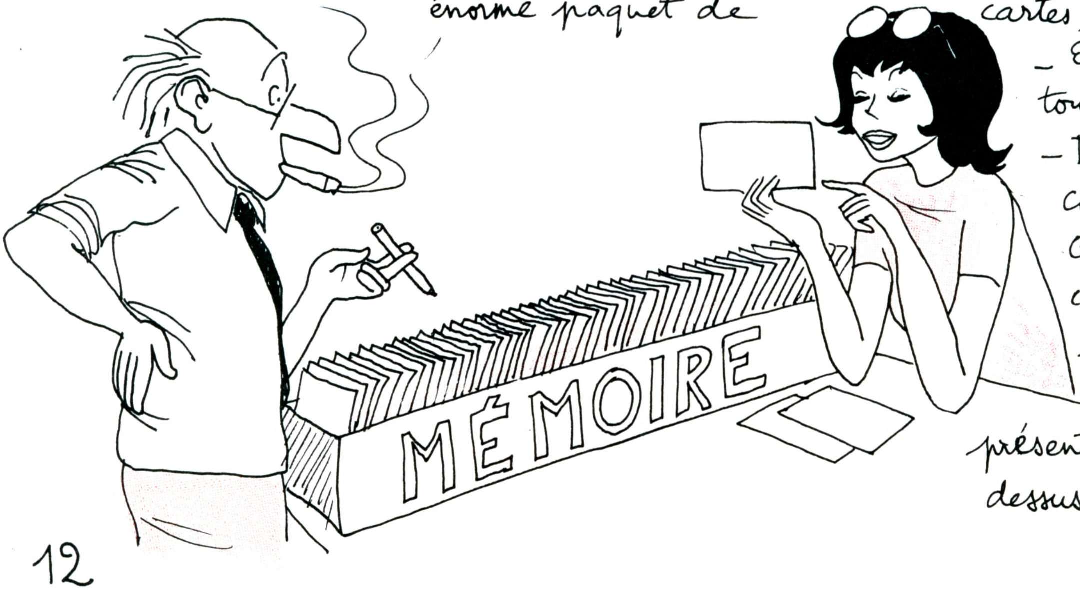
— Marie Thérèse, c'est affreux. J'ai acheté ce matériel. J'ai passé des heures à tout monter, à tout brancher. Et je ne suis pas plus avancé. Je suis complètement perdu dans ces centaines de pages de notices. Je ne sais même pas par où commencer.

— Pourquoi n'appelez vous pas votre nièce, Cerise. C'est la grosse tête de la famille.

— Vous êtes irremplaçable ! Pourquoi n'y avais-je pas pensé plus tôt !

MÉMOIRES

Oncle Achille, un ordinateur, au départ, c'est comme un énorme paquet de cartes, les CASES MÉMOIRE.



— Et que fait-on avec tout cela ?

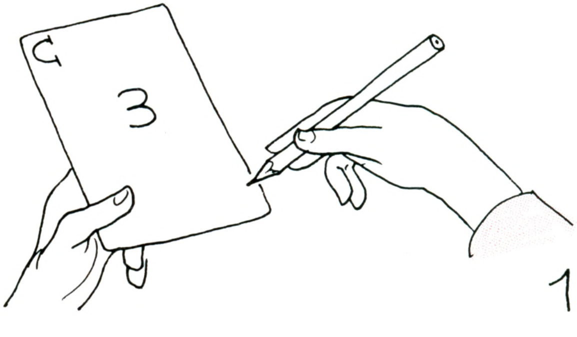
— D'abord, nous allons convenir d'un code. Quand je te dirai : « écris telle chose », tu prendras la première carte vierge qui se présentera et tu écriras dessus.

– Mais elles sont toutes vierges, s'écria Moneyback.
– Attends, on a pas encore commencé…
Et si je te dis "efface-tout !" alors tu gommeras ce que tu as mis.
– Bon ....
– Il nous faut maintenant un système de repérage pour ces cartes.
On appellera cela une **ADRESSE**. D'abord nous utiliserons les lettres de l'alphabet **A**, **B**, **C**, **D**, ... Nous pourrons aussi utiliser des groupes de deux lettres, ce qui nous fera 26 x 26 = 676 combinaisons possibles.
Ou encore une lettre suivie d'un chiffre, comme **A3** ou **Z6**.
Tout ceci fait partie d'un langage nommé **BASIC** et dont il sera question ici.

– Bon, dit Moneyback, je suis prêt, allons-y.
– Attention ! **C = 3**
Achille prit la première carte qui se présentait, inscrivit un C en haut et à gauche et un 3 au milieu. Puis il replaça la case dans le lot.
Cerise continua en annonçant :

B = 17

A = –8



– Nous avons réalisé une OPÉRATION D'ÉCRITURE DANS LA MÉMOIRE.
– Et comment s'effectue la LECTURE ?
– Je peux te demander "qu'y-a-t-il dans la case, la carte B ?"
– Je chercherai alors la carte qui est marquée d'un B et j'annoncerai son contenu.
– En BASIC cette lecture sera provoquée par l'ordre PRINT ou par l'ordre ? C'est à dire que je dirai PRINT B ou ?B indifféremment.

Achille se mit au clavier de l'ordinateur.

– Bon, alors pour écrire dans la mémoire je fais B = -17
– Ce n'est pas suffisant, il te faut spécifier la FIN DE MESSAGE. Il y a une touche pour cela. Suivant les ordinateurs elle portera l'inscription ENTER ou Return, que nous appellerons [Fin de Message].
– Et, pour interroger la mémoire, je ferai PRINT B, puis [Fin de Message].
– Exactement...
– Et que se passera-t-il si je fais PRINT Z [Fin de Message] alors que je n'ai pas créé de registre Z ?
– Dans ce cas là l'ordinateur affichera un zéro.
– Il est bizarre, ce zéro !
– En général le zéro des ordinateurs est barré Ø, pour le différencier de la lettre O.

Tu peux créer ainsi autant de registres que tu voudras, et décider de les annuler à l'aide de l'ordre, de l'INSTRUCTION CLEAR.

– Qu'appelles tu une INSTRUCTION.
– Une instruction, c'est une suite de caractères, de signes, qui ont un sens pour l'ordinateur.
– Un mot, quoi ?
– Si tu veux. Ces "mots" du LANGAGE de l'ordinateur s'articulent selon une SYNTAXE rigide.

MESSAGES D'ERREUR

Moneyback, pour s'entraîner, réalisa des opérations variées d'écriture et de lecture. Soudain la machine afficha un MESSAGE D'ERREUR. (Très souvent ce message est SYNTAX ERROR).

– Qu'est-ce que j'ai fait ? Fouchtre !...

– Tu as fait une ERREUR DE SYNTAXE. Tu as tapé ZM = 5,16 et il n'a pas aimé cela.

– Et alors ?

– Dans le langage des ordinateurs la virgule a une toute autre signification. Ici nous allons devoir suivre des règles anglosaxonnes, et la virgule des nombres décimaux devra être remplacée par un point.

– J'écrirai donc ZM = 5.16

Tout rentra dans l'ordre.

Mais quelques minutes plus tard, nouveau message d'erreur.

– Ici tu as tapé 30 au lieu de 3Ø, c'est classique.

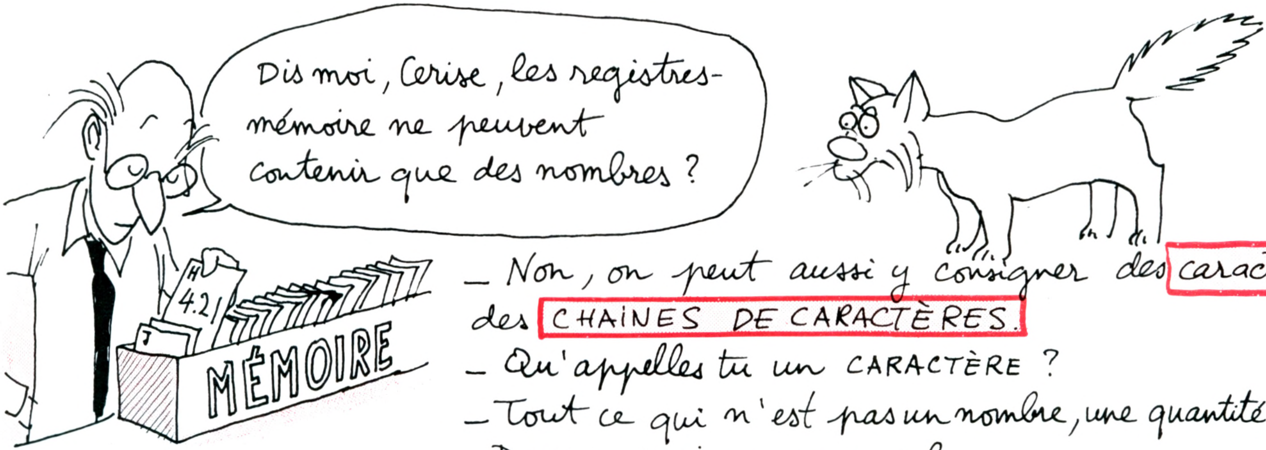
– Ah oui, ce zéro bizarre des informaticiens ! (*)

– Il faut bien un moyen quelconque pour différencier la lettre O du caractère représentant le zéro.

(*) IL FAUT BIEN DIFFÉRENCIER LA LETTRE O ET LE ZÉRO D'UNE MANIÈRE QUELCONQUE.

DANS LES MACHINES JAPONAISE, C'EST L'INVERSE : LE ZÉRO S'ÉCRIT O ET LA LETTRE O S'ÉCRIT Ø

CHAINES DE CARACTÈRES



— Non, on peut aussi y consigner des **caractères**, ou des **CHAINES DE CARACTÈRES.**

— Qu'appelles tu un CARACTÈRE ?

— Tout ce qui n'est pas un nombre, une quantité numérique.

— Donne-moi un escemple.

— N'importe quoi, des lettres, une astérisque.

— Alors il suffit de faire N = CERISE ?

L'ordinateur réagit aussitôt par un [MESSAGE D'ERREUR]

— Qu'est-ce qui ne lui plait pas ? demanda Moneyback.

— Toute écriture ou lecture de chaines de caractères devra comporter un **$**

— Que vient faire le dollar là-dedans ?

— Ce n'est pas le signe dollar $, cela vient de STRING, qui veut dire chaine en anglais.

— Bon, alors je suppose que l'on doit écrire N$ = CERISE

Nouveau message d'erreur...

— Il y a une syntaxe rigide. Tu dois mettre des guillemets et écrire :

```
N$="CERISE"
```

La lecture se faisant avec

```
PRINT N$
```

ROM ET RAM

Cerise revint au paquet de cartes.

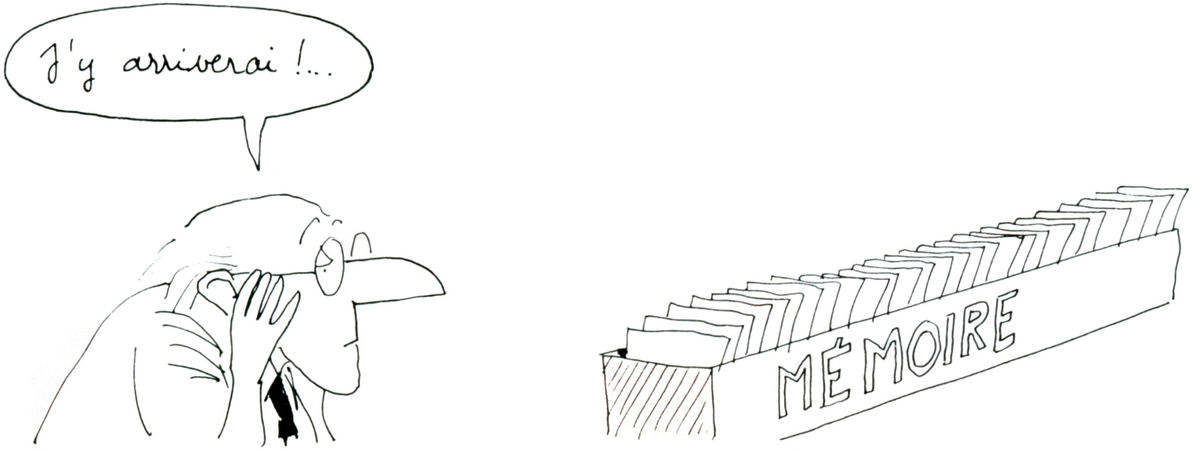
– L'ordinateur peut contenir beaucoup de choses variées. Certaines de ces choses sont effaçables et d'autres non. C'est à dire qu'il n'existe aucun ordre, aucune instruction, qui permette de modifier ces contenus.

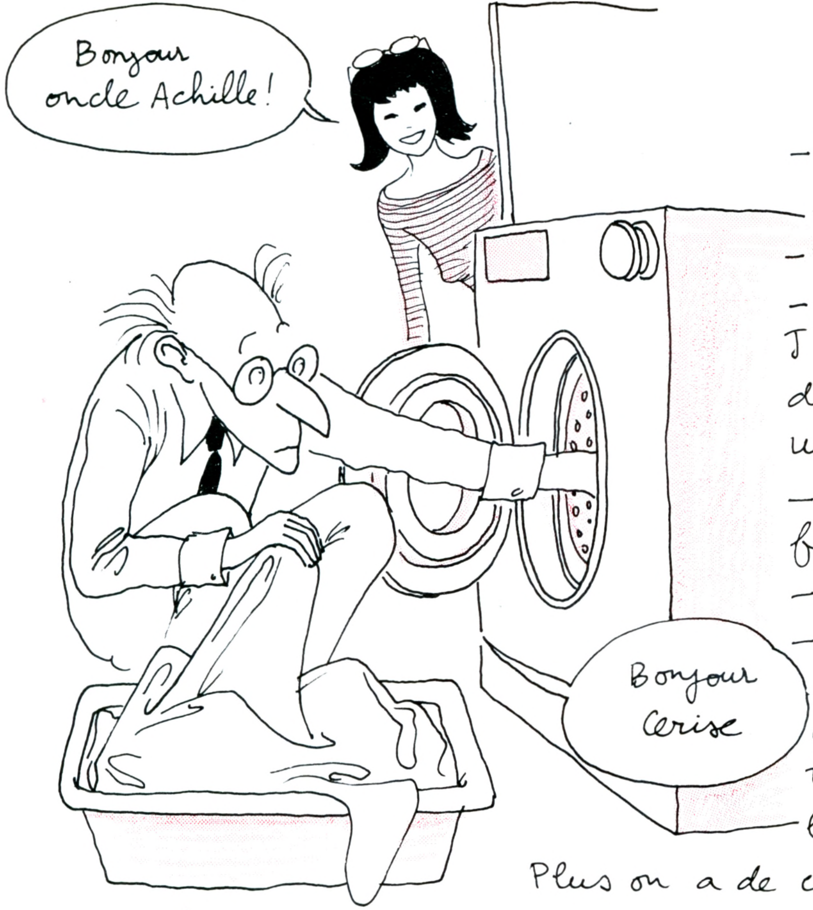
Ces éléments de mémoire où on ne peut écrire et qu'on ne peut effacer s'appellent des ROM (read only memory). Ce qui signifie en anglais "mémoire qu'on ne peut que lire".

Par opposition les mémoires ou l'utilisateur pourra écrire à sa guise s'appeleront RAM (random access memory) c'est à dire mémoire où l'on peut accéder à n'importe quel moment.

– Bon, dit Moneyback, je ne sais pas trop où tout cela me mènera, mais j'en ai assez appris pour aujourd'hui.

Cerise promit de revenir le lendemain. Achille resta longtemps songeur devant le classeur contenant le paquet de cartes.





— Tu en fais une tête !
— Je pensais à hier.
— Et bien ?
— Est-ce vraiment de l'informatique ? J'ai appris à consigner des choses sur des cases, et à les relier. C'est un peu limité, non ?
— Est-ce que tu es capable de faire cela en quelques microsecondes ?
— Euh.... non !
— Et bien ton ordinateur, oui ! Achille, nous ne faisons que commencer. Il ne faut pas te décourager. Il faut acquérir les connaissances petit à petit. Plus on a de connaissances, plus on sait faire de choses.
— Tu crois ?....
— J'en suis sûre ! Tiens, aujourd'hui je vais t'expliquer quelquechose de fondamental. Qu'est-ce que tu fais ?
— C'est le jour de congé de la femme de ménage. Et Marie-Thérèse m'a demandé de finir le linge.
Voyons ... que disent les PROGRAMMES de lavage.

1 — Très sale .... c'est pas ça
2 — sale
3 — Couleurs résistantes. Pas ça non plus
4 — Couleurs délicates. Ça doit aller...

LA NOTION DE PROGRAMME

— Bon. J'ai mis la lessive dans les casiers ad hoc.
C'est une machine perfectionnée. Quand j'ai choisi le type de programme un voyant s'allume et m'indique où il faut mettre la lessive. Puis je vois apparaître le message "mettez le linge dans le tambour". Si je n'ai pas bien refermé la porte, ma machine émet un son et me le rappelle. Quand tout est OK, elle démarre. Elle a neuf programmes de lavage.

— C'est comme l'ordinateur.

— Qu'est-ce que tu veux dire ?!?

— Lui aussi a des programmes.

— Ça par exemple ! tu veux dire qu'on peut lui faire laver le linge !!!

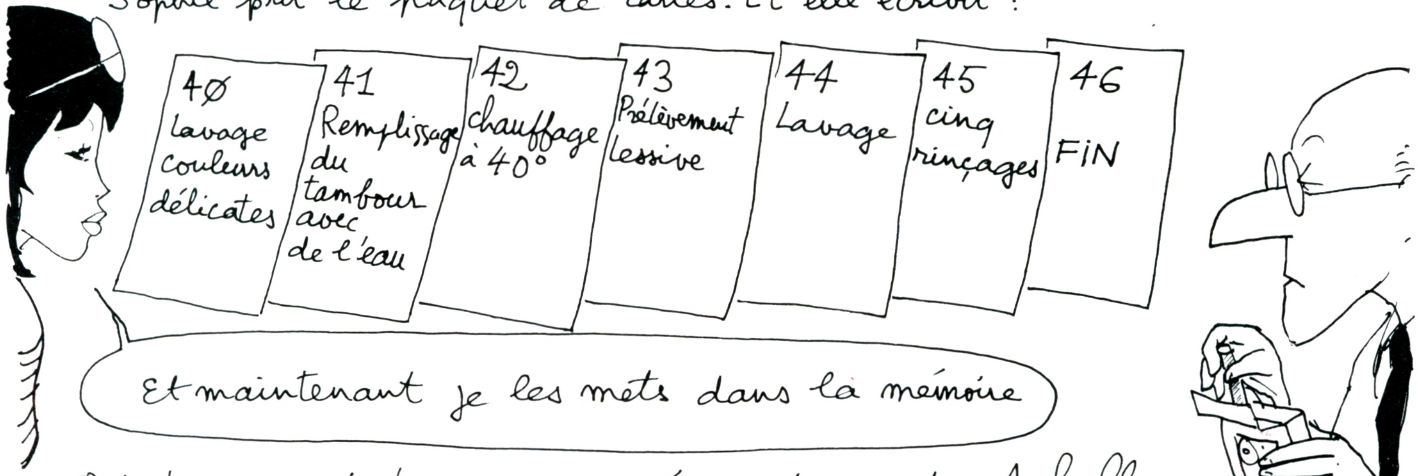
— Euh.... oui et non.... on peut lui faire faire beaucoup de choses.

Cerise prit un paquet de cartes dans le classeur.

— Voyons... quelles sont les opérations liées au programme de lavage. Cela doit être sur la notice. Ah... voilà ! programme couleurs délicates : Remplissage du tambour avec de l'eau, chauffage à 40°, prélèvement de la lessive disposée dans les casiers, lavage, cinq rinçages...

Il y a six programmes de lavage.

Sophie prit le paquet de cartes. Et elle écrivit :



– Qu'est-ce que c'est que ces numéros, demanda Achille.

– Ils ordonnent la séquences d'opérations, d'instructions, qui constituent ce PROGRAMME.

– Un programme est une SUITE ORDONNÉE D'INSTRUCTIONS ?

– Avec d'autres choses que nous verrons plus loin. Mais au départ c'est cela.

– Mais pourquoi as-tu commencé la numérotation à 40 ?

– J'aurais pu commencer par n'importe quel numéro. Et j'aurais pu numéroter les cartes de deux en deux, de dix en dix. Cela n'a aucune importance, seul l'ordre compte.

J'ai commencé par le chiffre 40 parce qu'il s'agissait du programme quatre.

– Je devine qu'entre 10 et 20 tu pourrais loger le programme numéro un, entre 20 et 30 le programme numéro trois, etc...

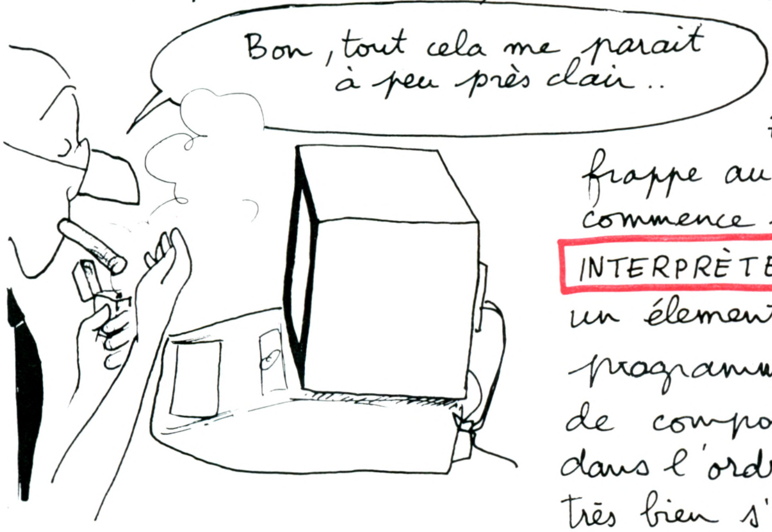
– Bien sûr, je peux mettre plusieurs programmes dans ma mémoire. Et je peux les faire exécuter à la demande.

– Comme moi avec ma machine à laver.

– comme tu as vu, à la fin, j'ai mis une [INSTRUCTION D'ARRÊT.]
Dans un ordinateur, si cette instruction n'est pas présente, la machine exécutera toute instruction repérée avec un numéro plus grand.

– Je suppose que dans la machine il doit y avoir quelquechose d'analogue.

– sinon, en lançant par exemple le programme deux, l'ordinateur, après exécution, enchainerait par le trois, le quatre, etc..



Résumons.

En langage BASIC, si je frappe au clavier quelquechose qui commence par un numéro, ceci sera INTERPRÈTÉ par l'ordinateur comme un élément (on dira une ligne) d'un programme. Il n'est pas nécessaire de composer ces LIGNES-PROGRAMME dans l'ordre. La machine saura très bien s'y retrouver

LE PREMIER PROGRAMME DE MONEYBACK

– Sophie, combien y a-t-il d'instructions dans un programme ?
– Autant que tu voudras, compte tenu de la capacité de la mémoire.
– Ce que je voulais dire, c'est : quel en est le nombre minimum ?
– Une. Tiens, tu peux par exemple imaginer un programme ne comportant qu'une seule instruction END [instruction d'arrêt], et qui s'écrirait :

1 END

Pour EXECUTER UN PROGRAMME, LANCER ou APPELER un PROGRAMME (expressions équivalentes), on doit composer RUN (et, bien sûr, appuyer sur la touche [Fin de message].
– Et que se passe-t-il ?
– L'ordinateur exécute e seul ordre présent qui est... de s'arrêter !
– Hum... il y a des gens qui, dans la vie, agissent ainsi.
– On peut continuer à s'amuser. Tiens, regarde cet autre programme.

1 RUN

– L'instruction RUN, dans certains ordinateurs, est PROGRAMMABLE. C'est à dire qu'on peut l'inclure dans un programme.
– Là, le gars se dit : « allons-y .... allons-y .... » à l'infini. Et en fait il ne fait rien !

– Soyons sérieux. Tu peux inclure dans un programme des ordres d'impression PRINT
– Comment procède-t-on ?
– La règle de syntaxe est la suivante : Tu inscris un PRINT dans ta LIGNE PROGRAMME. Puis tu ouvres les guillemets. Tu inscris ce que tu veux (sauf des guillemets), et tu n'oublies pas de refermer les guillemets.
– Sinon .... SYNTAX ERROR !

Achille composa :

```
1 PRINT "CLAVIER EST UN ANE"
```

Puis il lança ce programme à l'aide de RUN. La phrase s'inscrivit sur l'écran.

– Et, pour effacer ?
– Il existe une instruction [Effaçage Ecran], qui diffère d'un ordinateur à l'autre. On peut l'inclure dans un programme (ou l'utiliser directement)

```
1 [Effaçage Ecran]
2 PRINT "CLAVIER EST UN ANE"
```

– Comme cela il y a d'abord nettoyage de l'écran, puis impression de la phrase. Sophie, combien y a-t-il d'instructions ?
– L'ensemble des instructions constitue le LANGAGE de l'ordinateur. Il y en a évidemment pas mal. Apprendre à programmer, c'est se familiariser avec ce jeu d'instructions, à les combiner (SYNTAXE). La qualité d'un langage réside en particulier dans la variété et la souplesse de son jeu d'instructions.

Au cours de notre apprentissage, nous introduirons les INSTRUCTIONS au fur et à mesure des besoins. L'instruction NEW a pour effet de détruire, d'effacer tout programme dans la mémoire.

– Est-elle programmable ?

– Oui, ainsi tu peux créer des programmes qui, après utilisation, se détruisent eux-mêmes. Il suffit de créer en fin de programme une LIGNE portant l'ordre NEW (New veut dire nouveau, neuf)

```
1 [Effaçage Ecran]
2 PRINT "ATTENTION, CE PROGRAMME VA SE
DETRUIRE LUI-MEME !"
3 NEW
```

BUREAUTIQUE :

Moneyback composa NEW d'abord, au clavier, pour effacer toute trace du programme précédent (ce qu'il est conseillé de faire systématiquement, surtout sur les machines de poche qui peuvent conserver un programme en mémoire même quand elles sont éteintes)
Puis il écrivit d'une traite le programme ci-après :

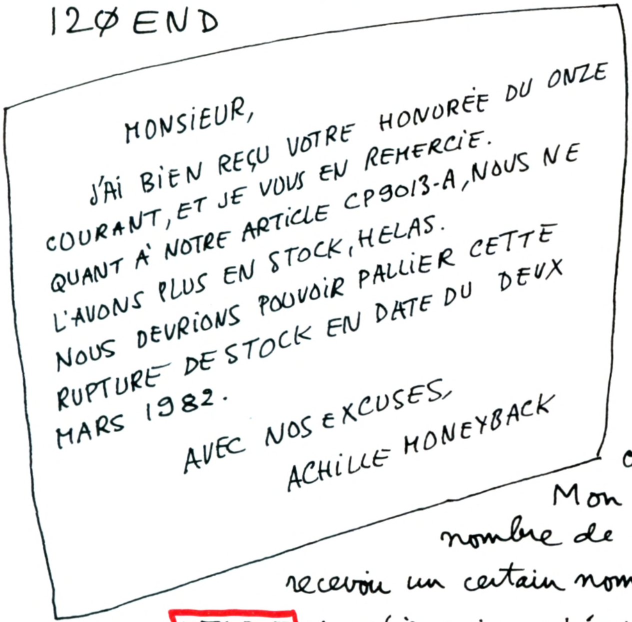
```
1 [Effaçage Ecran]
2 PRINT"MONSIEUR,"
3PRINT"J'AI BIEN RECU VOTRE HONOREE DU 11 COURANT"
4PRINT"ET JE VOUS EN REMERCIE"
5PRINT"QUANT A NOTRE ARTICLE CP9013-A, NOUS NE"
6PRINT"L'AVONS PLUS EN STOCK, HELAS."
7PRINT"NOUS DEVRIONS POUVOIR PALLIER CETTE"
8PRINT"RUPTURE DE STOCK EN DATE DU 2"
9PRINT"MARS 1982"
10PRINT"AVEC NOS EXCUSES "
11PRINT"          ACHILLE MONEYBACK"
12END
```

Moneyback fit RUN, et vit apparaître :

Sophie se mit au clavier :

– Attends, on va arranger cela. Elle fit NEW, et composa à son tour :

```
10 [Effaçage Ecran]
20 VTAB5 : HTAB16 : PRINT"MONSIEUR,"
30 PRINT:PRINT"J'AI BIEN RECU VOTRE HONOREE DU ONZE"
40 PRINT:PRINT"COURANT, ET JE VOUS EN REMERCIE" : PRINT
50 PRINT "QUANT A NOTRE ARTICLE 9013-A, NOUS NE"
60 PRINT: PRINT"L'AVONS PLUS EN STOCK, HELAS."
70 PRINT : PRINT" NOUS DEVRIONS PALLIER CETTE"
80 PRINT: PRINT" RUPTURE DE STOCK EN DATE DU DEUX"
90 PRINT: PRINT"MARS 1982" : PRINT
100 HTAB8: PRINT" AVEC NOS EXCUSES"
110 PRINT: HTAB17 : PRINT "ACHILLE MONEYBACK"
120 END
```



A la ligne 20 j'ai mis des INSTRUCTIONS DE TABULATION. Elles peuvent différer d'un ordinateur à l'autre, mais en voici le principe : Mon écran d'affichage comprend un certain nombre de lignes. Et chaque ligne est susceptible de recevoir un certain nombre de CARACTÈRES (y compris les blancs).

VTAB5 signifie qu'on n'écrira qu'à partir de la 5e ligne. Et HTAB16 veut dire : n'écrire qu'à partir de la seizième case de la ligne.

– Et les deux points : ?

– Ah, c'est un truc qui permet de loger deux instructions sur la même ligne.

– Et le PRINT en début de ligne 3Ø ?

– Il donne une ligne vide, un blanc, à l'affichage. C'est pour la présentation. Sinon le texte est un peu serré.

– Est-ce que j'ai un moyen de repérer un programme, de lui donner un nom, par exemple, qui me permette de m'y retrouver ?

– Tu peux consigner des REMARQUES dans un programme BASIC, à l'aide de l'instruction **REM**

– Mais comment allons nous MODIFIER ce programme maintenant qu'il est composé ? Il va falloir tout reprendre à zéro ?

– Pas du tout ! Comme tu vois j'ai numéroté les lignes de dix en dix ce qui me laisse de la place pour en rajouter. Il me suffit de taper :

```
5 REM  LETTRE SUR RUPTURE STOCK
7 REM  ARTICLE  CP 9013-A
```

– Mais, comment peux-tu vérifier que ces deux lignes ont bien été incluses dans le programme ?

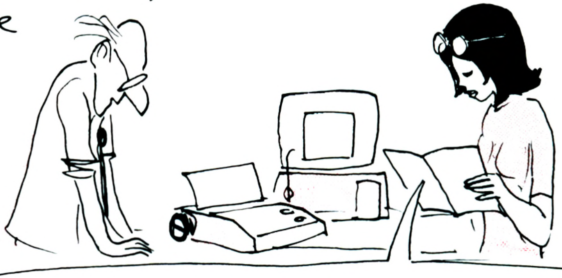
– A tout moment, je peux provoquer l'affichage, sur l'écran, du programme en utilisant l'ordre, la commande

LIST

Mais je peux "LISTER" une ligne particulière en faisant **LIST 3Ø**
Ou même une suite de lignes en composant :

LIST 5,8Ø

(lignes 5 à 8Ø, incluses)



– Si je fais une erreur en composant une ligne, qu'est ce que je dois faire ?

– Sur le clavier, tu as d'abord deux touches [←] et [→]. La première permet un retour en arrière. Ce qui te permet de corriger caractère par caractère. En utilisant la barre d'espacement, tu imprimes un BLANC, un caractère "VIDE", ce qui permet l'effacement.

Pour annuler une ligne, le plus simple est de composer son numéro puis de presser la touche [FIN DE MESSAGE]. Pour remplacer une ligne par une autre, cette opération préalable n'est pas nécessaire. Il suffit de réécrire cette ligne, puis de presser la touche [FIN DE MESSAGE].

Pour effacer une suite de lignes, tu utilises [DEL 5Ø, 11Ø] (ceci signifie : annuler les lignes 5Ø à 11Ø, incluses).

– Dis moi, en utilisant astucieusement les instructions HTAB et VTAB je peux placer sur une feuille l'adresse de mon correspondant, au bon endroit, pour qu'elle puisse se lire à travers la fenêtre transparente d'une enveloppe.

– Si tu veux.

Cerise partie, Moneyback resta longtemps pensif.

Moneyback se mit au clavier, se trompa de nombreuses fois, corrigea ses erreurs. Quand Marie-Thérèse rentra, il lui dit :

– Ma douce, viens voir.

– Voir cette affreuse machine !

– Viens, je t'en prie, pour me faire plaisir.

– Bon...

– Maintenant, composes RUN et presse sur cette touche [FIN DE MESSAGE]

La machine afficha aussitôt :

MA BICHETTE,

VEUX TU QUE JE T'EMMENE AU RESTAURANT CE SOIR ?

TON GROS LAPIN.

PROGRAMMES AVEC ENTRÉES DE DONNÉES

Cerise trouva son oncle très perplexe.

— J'ai eu une fausse joie hier. Je croyais que tout cela allait pouvoir me servir à quelquechose. Mais, regarde ce programme qui écrit des lettres. Je ne peux même pas modifier le numéro de l'article.

Il faut, continua Cerise, spécifier quel type de REGISTRE l'ordinateur devra créer pour mettre cette donnée.

INPUT N mettra l'ordinateur en attente d'une donnée numérique.

INPUT N$ mettra l'ordinateur en attente d'une chaine de caractères.

Et tu peux aussi faire poser, en clair, la question par la machine.

Exemple :

```
20 INPUT "DATE LETTRE CLIENT :"; D$
```

Attention ⇒ n'oublier ni les guillemets, ni le point virgule.

(*) INPUT : en anglais, signifie ENTRÉE

– Et on peut tout mettre dans les guillemets ?
– Sauf des guillemets ....

Moneyback modifia le programme précédent, comme suit :

```
5 REM LETTRE SUR RUPTURE DE STOCKS
10 [Effaçage Ecran]
20 INPUT"DATE LETTRE CLIENT: ";D$
30 INPUT"REFERENCE NOTRE ARTICLE: "; AR$
40 INPUT"DATE NOTRE REPONSE: ";R$
50 PRINT"DATE A LAQUELLE LE CLIENT POURRA ETRE"
55 PRINT
60 INPUT"LIVRE:";LI$
70 HOME
80 [Activation de l'Imprimante]
90 PRINT"SOCIETE MONEYBACK"
100 HTAB 20 : PRINT"LE "; R$
110 VTAB5 : HTAB 16 : PRINT "MONSIEUR,"
120 PRINT:PRINT"BIEN RECU VOTRE LETTRE DU ";D$ : PRINT
130 PRINT "DONT NOUS VOUS REMERCIONS." : PRINT
140 PRINT"QUANT A L'ARTICLE ";AR$;" ,NOUS NE " :PRINT
150 PRINT"L'AVONS PLUS EN STOCK,HELAS.": PRINT
160 HTAB11:PRINT"NOUS POURRIONS PALLIER CETTE":PRINT
170 PRINT"RUPTURE DE STOCK VERS LE ";LI$ :PRINT
180 HTAB8 : PRINT"AVEC NOS EXCUSES," : PRINT
190 HTAB17 : PRINT"ACHILLE MONEYBACK"
200 [DESACTIVATION DE L'IMPRIMANTE]
210 END
```

Plusieurs remarques. A la ligne 140 le point virgule permet d'enchaîner des impressions (sur l'écran et sur le papier), que les choses à imprimer soient inscrites directement dans le programme ou présentes dans des registres (ici AR$). Le PRINT de séparation de ligne peut être mis indifféremment en début ou en fin de ligne.

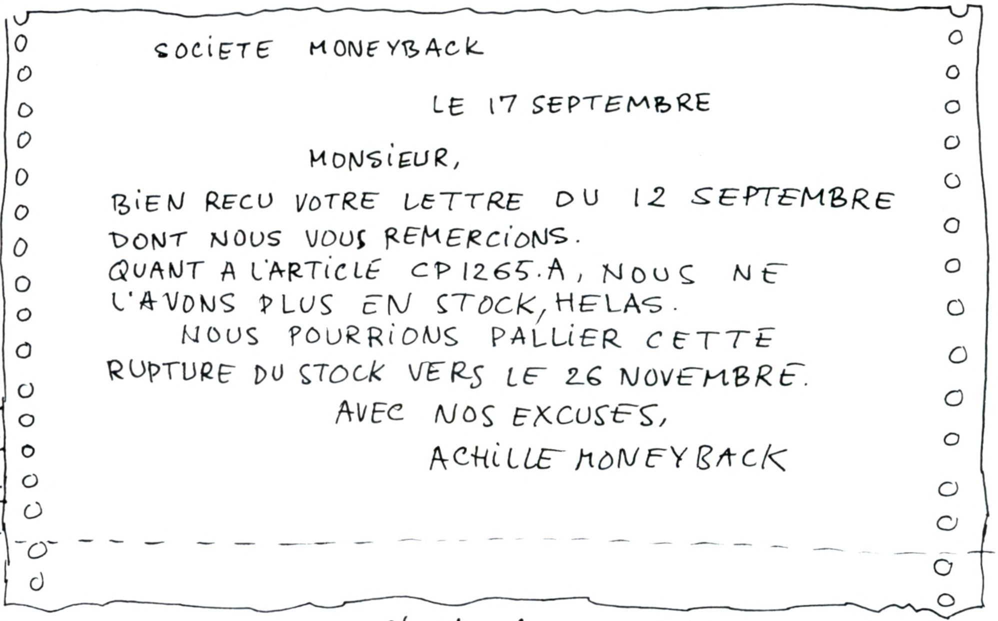
Si on ne dispose pas des instructions VTAB et HTAB on peut remplacer la première par une série de PRINT : PRINT : PRINT (chaque PRINT entraîne un saut de ligne), et la seconde par des "

Moneyback fit RUN et sur l'écran apparut :

DATE LETTRE CLIENT ▨
curseur clignotant

– Bon, qu'est-ce que je fais ?
– Bien, tu tapes la date, par exemple 12 SEPTEMBRE. Puis tu appuies sur la touche [FIN DE MESSAGE].

Moneyback répondit ainsi aux différentes questions, et la machine imprima :


SOCIETE MONEYBACK

LE 17 SEPTEMBRE

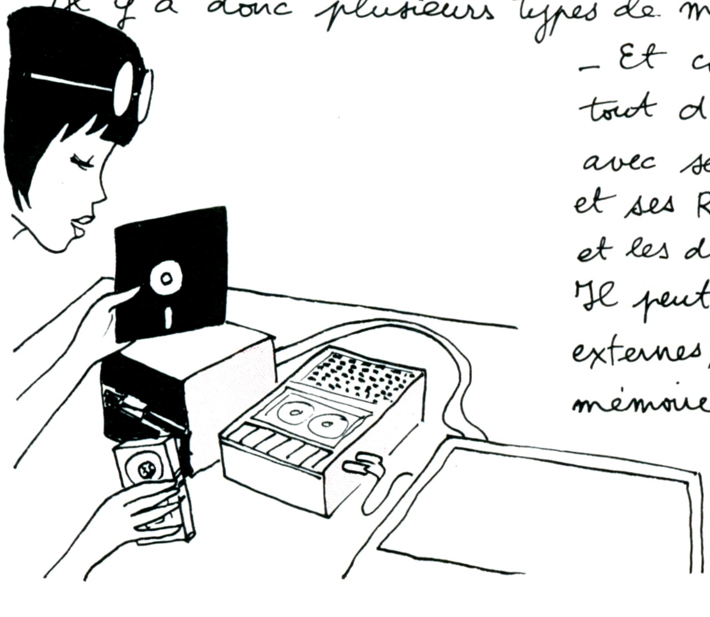
MONSIEUR,

BIEN RECU VOTRE LETTRE DU 12 SEPTEMBRE
DONT NOUS VOUS REMERCIONS.
QUANT A L'ARTICLE CP1265.A, NOUS NE
L'AVONS PLUS EN STOCK, HELAS.
NOUS POURRIONS PALLIER CETTE
RUPTURE DU STOCK VERS LE 26 NOVEMBRE.
AVEC NOS EXCUSES,
ACHILLE MONEYBACK

A l'aide de ce premier programme Achille venait d'entrer de plein pied dans la jungle de l'informatique.

STOCKER, CHARGER UN PROGRAMME

– Mon ordinateur est donc prêt à m'écrire autant de lettres que je le veux.
– Tu mettras le premier type de lettre de la ligne 5 à la ligne 210 puis la seconde de la ligne 1000 à la ligne 1320 par exemple. La troisième de la ligne 2000 à .... etc. Et tu "appelleras" ces différents programmes en faisant RUN5 ou RUN1000 ou RUN 2000, etc...
– Tous les problèmes sont résolus...
– Tu crois ? Ton ordinateur est immobilisé. Il ne peut plus rien faire d'autre. Et si quelqu'un, en ton absence, éteint la machine, ou que survienne une coupure de courant : Hein, envolé, ton programme (à moins que ton ordinateur n'ait une MÉMOIRE NON VOLATILE.)
Il va te falloir stocker, RECOPIER, ce programme dans une MÉMOIRE EXTERNE.
Il y a donc plusieurs types de mémoires ?



– Et comment ! Notre micro ordinateur a tout d'abord sa MÉMOIRE CENTRALE avec ses ROM (son LOGICIEL DE BASE). et ses RAM, où s'inscrivent les programmes et les données.
Il peut être relié à deux types de mémoires externes, une mémoire à cassette et une mémoire à DISKETTES, plus commode.

– Commençons par le magnétophone. Le système manque de souplesse. Il faut d'abord positionner la bande, en s'aidant du compteur. Ne serait-ce que pour trouver une place vide sur celle-ci.
Un ordre clavier (CSAVE) provoque la recopie du programme, codé en binaire, qui se trouve transféré sur la bande sous forme d'une suite d'impulsions.
La recherche du programme est laborieuse. On se positionne à l'aide du compteur du magnétophone. Sur certains ordinateurs l'ordre de stockage est simplement CSAVE et l'ordre de chargement CLOAD (voir notice).
Le CHARGEMENT est une opération où ce qui est présent sur la bande va se trouver recopié en mémoire centrale. Cette recopie n'altère en rien le contenu de cette mémoire périphérique. Même remarque pour l'opération de STOCKAGE (qui n'altère pas ce qui est en mémoire centrale).
Certains systèmes permettent de repérer le programme à l'aide d'un IDENTIFICATEUR de quelques lettres

Exemple : CSAVE "LETTRES" (*)
CLOAD "LETTRES"

Dans ces conditions, en explorant la bande, l'ordinateur est capable de trouver lui-même le programme recherché.
Mais on reste tributaire de la vitesse, assez lente, de défilement de la bande ....

(*) SAVE, en Anglais, veut dire SAUVER, préserver.
LOAD veut dire CHARGER

Le système à disques souples, DISKETTES, est infiniment plus commode, rapide et souple.

Les ordres "diskettes" sont :

SAVE PROGRAMME LETTRES

LOAD PROGRAMME LETTRES

Sans guillemets.

Lorsqu'on compose CATALOG, la machine donne immédiatement les programmes contenus sur cette face de la diskette.

Dans les jours qui suivirent Achille imagina tous les types de lettres possibles, qu'il stocka en mémoire périphérique, soit sur la minicassette, soit sur des diskettes.

misère...

BOUCLES

Bon. La bureautique, ça va à peu près. Il faut que je mette un peu d'ordre dans toutes ces procédures de réponse. Tout doit être prévu à l'avance...

Pfff.... une heure et quart. c'est l'insomnie !
Allô, Cerise, excuse-moi de t'appeler si tard. Je n'arrive pas à dormir. C'est peut être la conjecture ?..
Ecoute, oncle Achille, je vais te dicter un programme qui devrait t'aider. Tu n'as qu'à noter...
Alors, tu dis :
5 [Effaçage Ecran]
10 PRINT "1 MOUTON"
20 N = 1
30 N = N + 1
40 PRINT N ; "MOUTONS"
50 GOTO 30
N = N + 1
cela semble absurde !.. enfin ....

Le lendemain Cerise donna quelques explications sur ce programme :

– Ce programme "compte" les moutons – ligne 5 : nettoyage de l'écran. Ligne 10 : on imprime d'abord "1 MOUTON", au singulier. Ligne 20 apparaît le registre N qui constitue notre **COMPTEUR**. Valeur initiale : un.

Ligne 30 : une opération fondamentale de l'informatique : **L'INCRÉMENTATION**

– Parce que N = N + 1 cela ne veut rien dire. Cela signifierait : un égale zéro !

– Dans un programme, le signe = n'a pas le sens qu'on lui attribue en algèbre. Quand j'écris N =, cela signifie : mettre dans le registre N telle chose. Ainsi **N = N + 1** doit se lire : mettre dans le registre mémoire N la valeur qui était contenue dans ce registre, accrue d'une unité. On devrait écrire N ⇐ N + 1.

– Alors, incrémenter, c'est quand on accroît un registre d'une unité ?

– Non, tu peux incrémenter d'une quantité quelconque, entière ou décimale, qu'on appellera **L'INCRÉMENT**. Tu peux aussi **DÉCRÉMENTER**. Par exemple **I = I – 3.601**

– A la ligne 40 tu retrouves un ordre d'impression qui affiche bout à bout N, c'est à dire le nombre de moutons, suivi de "MOUTONS". Et en 50 tu as une instruction de BRANCHEMENT INCONDITIONNEL, qui te renvoit à la ligne 30. GOTO, en anglais, veut dire "aller vers". Ce que faisant, on a créé une BOUCLE SANS FIN.

– Effectivement, on est déja à plus de cent mille moutons. Comment fait-on pour stopper un tel programme?

– Cela dépend des machines. Il existe une [INSTRUCTION D'ARRET] que l'on compose au clavier. L'ordinateur alors s'arrète net, et affiche la dernière ligne exécutée.

– Dis moi, Cerise, tout à l'heure, ça allait beaucoup trop vite. Au moins une ligne imprimée sur l'écran à chaque seconde. On pouvait à peine lire. Peut-on ralentir cela?

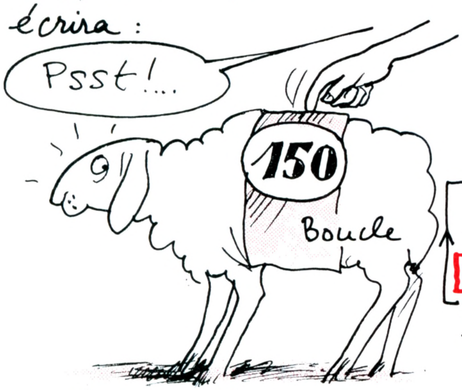
– Oui, mais je te montrerai cela plus tard.

– Mais je vais te montrer comment on peut compter un nombre donné de moutons. Pour cela il faut utiliser une **INSTRUCTION DE BRANCHEMENT CONDITIONNEL**

– Montre moi cela !

BRANCHEMENTS CONDITIONNELS

– On utilise l'instruction IF... THEN... (SI.... ALORS...)
Supposons qu'on veuille compter cent cinquante moutons. On écrira :




```
5 [Effaçage Ecran]
1Ø PRINT "1 MOUTON"   (*)
2Ø N=1
3Ø N=N+1
4Ø PRINT N ; " MOUTONS"
5Ø IF N=15Ø THEN 7Ø
6Ø GOTO 3Ø
7Ø END
```

Sortie de boucle.

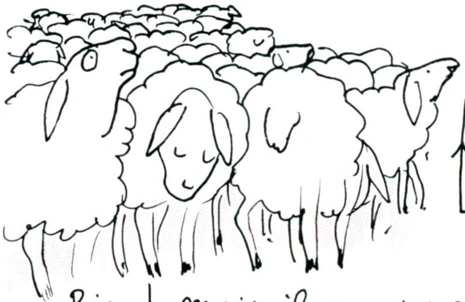
A chaque fois que l'on repassera par la ligne 5Ø un TEST sera effectué. Tant que N est different de 15Ø le bouclage continuera. Mais si N = 15Ø alors il est prescrit d'aller en ligne 7Ø, et on réalise une **SORTIE DE BOUCLE.** En 7Ø on a mis une instruction d'arrêt END. Alors le travail s'arrête.

– Ah, dit Achille, j'ai une autre idée. On peut utiliser une instruction INPUT pour préciser, avant exécution, combien de cycles sont à effectuer

– Vas-y

(*) Sur les machines de poche une instruction PAUSE affiche pendant environ une seconde les caractères entre guillemets

– Moneyback composa le programme suivant :



```
5 [Effaçage Ecran]
7 INPUT"COMBIEN DE MOUTONS ?";NM
9 [Effaçage Ecran]
10 PRINT" 1 MOUTON"
20 N = 1
30 N = N+1
40 PRINT N; " MOUTONS"
50 IF N= NM THEN 70
60 GOTO 30
70 END
```

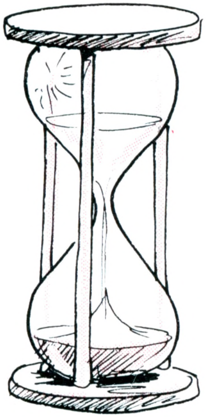
– Bien ! mais il y a une façon plus compacte d'écrire cela :

BOUCLES FOR... NEXT

```
5 [ Effaçage Ecran]
7 INPUT"COMBIEN DE MOUTONS ?";NM
9 [ Effaçage Ecran]
10 PRINT " 1 MOUTON"
20 FOR N = 2 TO NM
30 PRINT N; " MOUTONS"
40 NEXT N
50 END
```

– Tu vois, ça se fait tout seul. La ligne 2ø signifie "pour N allant de 2 à NM". La ligne 4ø signifie "Valeur suivante de N". Et maintenant je vais pouvoir te montrer comment on peut régler le temps d'affichage non permanent sur l'écran.

BOUCLES DE TEMPORISATION



L'informatique permet de simuler un comportement, très courant, qui consiste à refaire la même chose éternellement. En psychanalyse ceci s'appelle la compulsion de répétition.

L'ordinateur peut rééditer éternellement une tâche sans aucun contenu. Par exemple, à travers le programme :

```
1 GOTO 1
```

Pour "temporiser" on introduira à l'endroit voulu le nombre ad hoc de cycles vides. On peut loger la boucle sur une seule ligne. Par exemple, dans le programme précédent, on pourrait caser :

```
35 FOR I=Ø TO 5ØØ : NEXT I
```

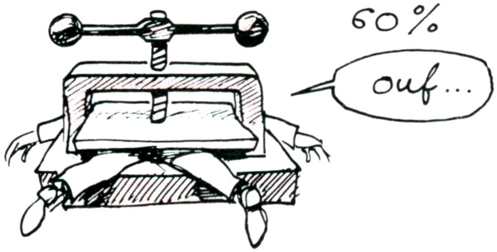
Le délai introduit est évidemment proportionnel au nombre de boucles.

Moneyback se demanda si, en utilisant ce qu'il savait déja, il ne pourrait pas mettre sous forme d'un programme le calcul de l'impôt sur le revenu.

– Voyons, on part du revenu imposable R et du nombre de parts N. Deux informations qui pourront être données à la machine grâce à des instructions INPUT.

On divise R par N et on obtient ainsi le quotient familial QF.
Le calcul de l'impôt dépend alors de la valeur de QF. Sur la feuille de l'administration, Achille lut :

Votre quotient QF :	T taux de l'impôt	Calcul de l'Impôt I
Si QF < ou = à 9890F	0%	I = 0
Si QF > 9890F et QF <= 10340F	5%	I = (R×0,05) – (494,5 ×N)
Si QF > 10340F et QF <= 12270F	10%	I = (R×0,10) – (1011,5 ×N)
Si QF > 12270F et QF < 19410F	15%	I = (R×0,15) – (1625 ×N)
Si QF > 19410F et QF <= 24950F	20%	I = (R×0,20) – (2595,5 ×N)
Si QF > 24950F et QF <= 31360F	25%	I = (R×0,25) – (3843 ×N)
Si QF > 31360F et QF <= 37940F	30%	I = (R×0,30) – (5411 ×N)
Si QF > 37940F et QF <= 43770F	35%	I = (R×0,35) – (7308 ×N)
Si QF > 43770F et QF <= 72940F	40%	I = (R×0,40) – (9496,5 ×N)
Si QF > 72940F et QF <= 100320F	45%	I = (R×0,45) – (13143 ×N)
Si QF > 100320F et QF <= 118660F	50%	I = (R×0,5) – (18159,5 ×N)
Si QF > 118660F et QF <= 135 000F	55%	I = (R×0,55) – (24092,5 ×N)
Si QF > 135 000F	60%	I = (R×0,6) – (30842,5 ×N)



CALCULER SES IMPOTS

Bon, se dit Achille, chaque ligne de ce tableau figure une TRANCHE D'IMPOSITION
Le début n'est pas difficile. Je n'ai qu'à mettre :

```
1Ø [Effaçage Ecran]
2Ø PRINT"CALCUL D'IMPOT SUR LE REVENU"
3Ø PRINT
4Ø INPUT"VOTRE REVENU IMPOSABLE :";R : PRINT
5Ø INPUT"NOMBRE DE PARTS :";N : PRINT
6Ø QF=R/N : REM QUOTIENT FAMILIAL
7Ø PRINT"VOTRE QUOTIENT FAMILIAL :";QF
```

Pour la suite, je devrais pouvoir m'en tirer avec une série de ....IF.... THEN....

Cerise vint à la rescousse :
– Formidable, mon oncle, tu fais des progrès ! Je vais t'aider un peu.
L'instruction IF.... THEN... se base sur un test, mais tu peux donner à ce test plusieurs facettes, en utilisant le "ET" et le "OU", ce qui te donnera SI CECI ..... ET CELA .... ET CELA ... ALORS ....
ou : SI CECI .... OU CELA ..... ALORS ....
"ET" en anglais se dit AND
"OU" se dit "OR"
Dans ces conditions, voilà comment s'écrirait par exemple la deuxième ligne du tableau du calcul de l'impôt :

```
IF QF>989Ø AND QF<=1Ø34Ø THEN T=5 : I=R*Ø.Ø5-494.5*N
```

– Fouchtre ! tu as réussi à tout faire tenir sur une seule ligne.

– A gauche, tu as les deux TESTS, "supérieur à 989Ø" et "inférieur ou égal à 1Ø . Et, à droite, tu as les différentes actions qui en découlent, séparées par les deux points :

– Alors le THEN (ALORS) peut porter sur tout ce qui suit ?

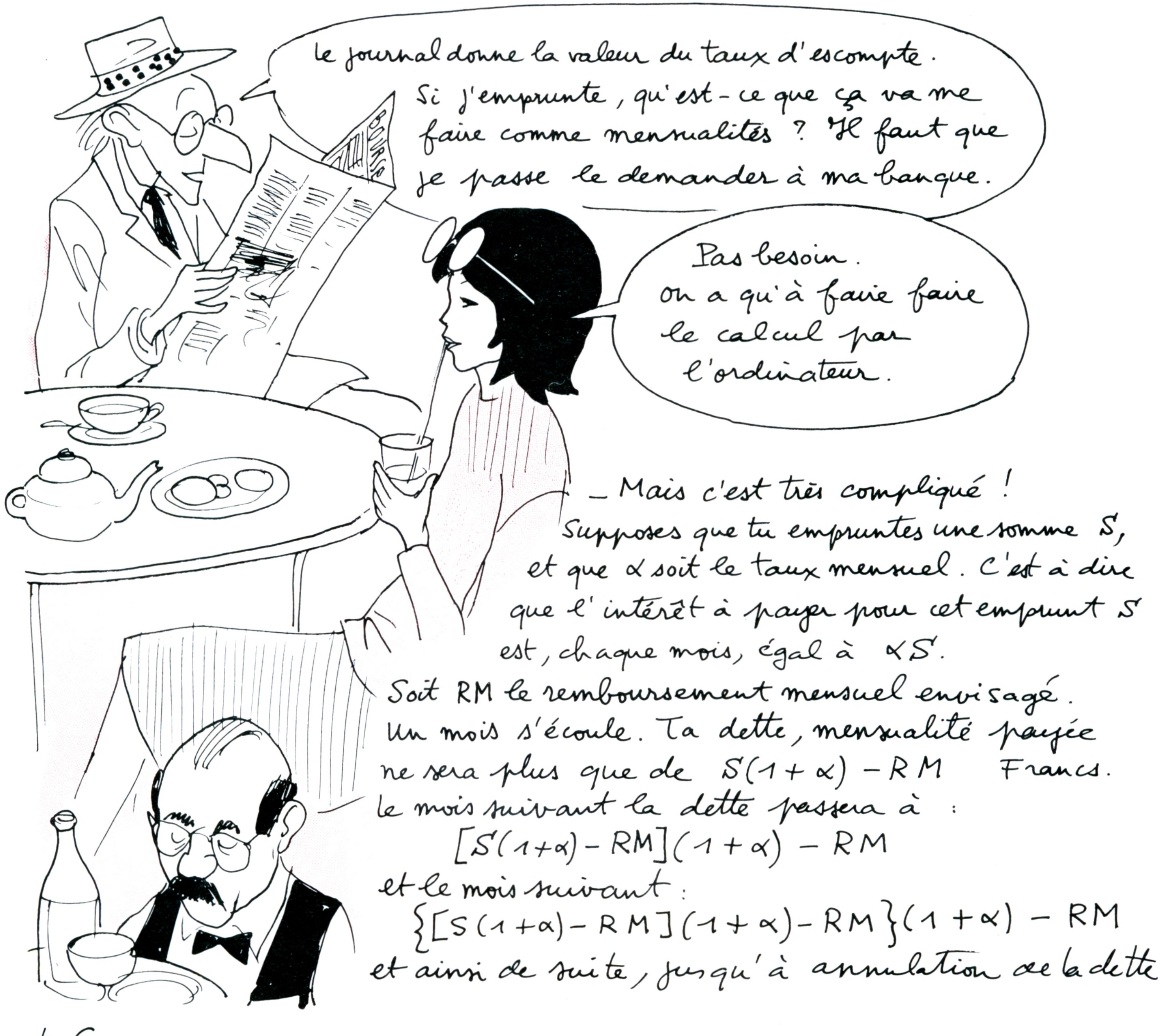
– Oui, Ou tu mets un numéro de ligne (branchement conditionnel), ou tu mets la suite d'opérations à effectuer si le test est satisfait (*)

Alors le programme complet s'écrira :

```
1Ø   PRINT"CALCUL D'IMPOT SUR LE REVENU": PRINT
2ØINPUT"DONNEZ VOTRE REVENU IMPOSABLE :";R : PRINT
3ØINPUT"DONNEZ LE NOMBRE DE PARTS : ";N : PRINT
4Ø QF=R/N :PRINT"VOTRE QUOTIENT FAMILIAL EST : ";QF : PRINT
5Ø IF QF<=989Ø THEN T=Ø : I=Ø : GOTO 18Ø
6Ø IF QF>989Ø AND QF<=1Ø34Ø THEN T=5 : I=R*Ø.Ø5-494.5*N : GOTO 18Ø
7Ø IF QF>1Ø34Ø AND QF<=1227Ø THEN T=1Ø : I=R*Ø.1-1Ø11.5*N:GOTO 18Ø
8Ø IF QF>1227ØAND QF<=1941Ø THEN T=15 : I=R*Ø.15-1625*N : GOTO 18Ø
9Ø IF QF>1941ØAND QF<=2495Ø THEN T=2Ø : I=R*Ø.2Ø-2595.5*N : GOTO 18Ø
1ØØ IF QF>2496ØAND QF<=3136Ø THEN T=25 : I=R*Ø.25-3843*N : GOTO 18Ø
11Ø IF QF>3136ØAND QF<= 3794Ø THEN T=3Ø : I=R*Ø.3Ø-5411 *N : GOTO 18Ø
12Ø IF QF>3794ØAND QF <=4377Ø THEN T=35 : I=R*Ø.35-73Ø8 *N : GOTO 18Ø
13Ø IF QF>4377ØAND QF<=7294Ø THEN T=4Ø : I=R*Ø.4-9496.5*N :GOTO 18Ø
14Ø IF QF>7294ØAND QF<=1ØØ32Ø THEN T=45 : I=R*Ø.45-13143.5*N :GOTO 18Ø
15Ø IF QF>1ØØ32Ø AND QF<=11866Ø THEN T=5Ø : I=R*Ø.5-18159.5*N :GOTO 18Ø
16Ø IF QF>11866Ø AND QF<=135ØØØ THEN T=55 : I=R*Ø.55-24Ø92.5*N :GOTO 18Ø
17Ø IF QF>135ØØØ THEN T=6Ø : I=R*Ø.6-3Ø842.5 *N
18Ø PRINT"VOTRE TAUX D'IMPOSITION EST :";T;" POUR CENT" : PRINT
19ØPRINT"MONTANT DE VOTRE IMPOT SUR LE REVENU:";I;" F"
2ØØEND
```

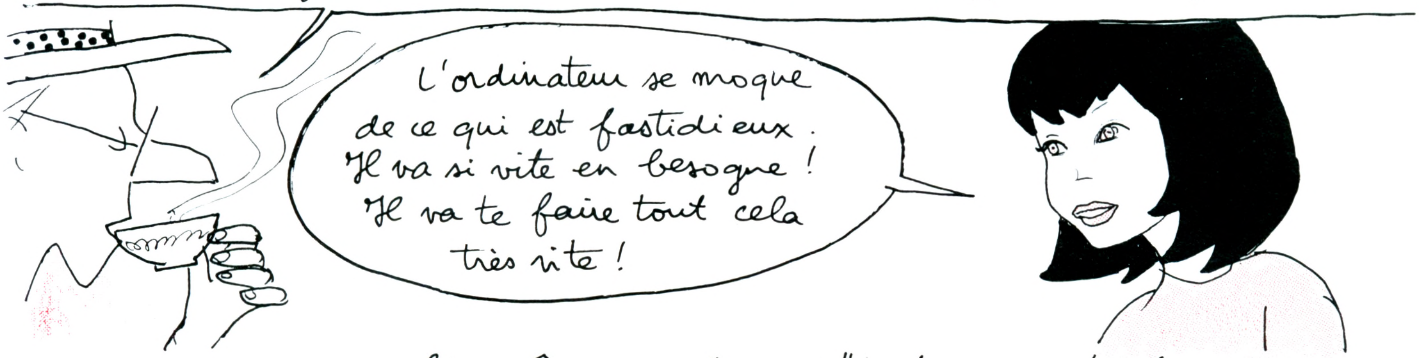
* On notera que les signes de l'arithmétique sont (addition +) (division /) (soustraction –) (multiplication *)

EMPRUNTER DE L'ARGENT



– Mais si ton emprunt tourne sur des dizaines de mois, ta formule devient plutôt longue !

– Attends ! la question que tu poses est : " si je rembourse en NA années quelle sera ma mensualité RM ?". De fait, ce calcul donne l'INVERSE : Quand on choisit le remboursement mensuel RM, il fournit par récursivité, le temps nécessaire pour éponger la dette. En se servant de cette "formule" il faudra donc faire un grand nombre de fois ce calcul, en essayant diverses valeurs de RM, jusqu'à ce qu'on tombe pile, ou suffisamment près.



On va commencer, bien sûr, par "ENTRER" tout ce qui est nécessaire.

```
10 [Effaçage Ecran]
20 INPUT "TAUX D'ESCOMPTE, EN %";TE
30 A=TE/1200 : PRINT
40 INPUT "SOMME EMPRUNTEE:";S : PRINT
50 INPUT "SUR COMBIEN D'ANNEES ?";NA
```

A est le TAUX MENSUEL. On a conjugué une division par cent (%) et par douze (les mois de l'année).

```
60 NM=NA*12 :REM NOMBRE DE MOIS
```

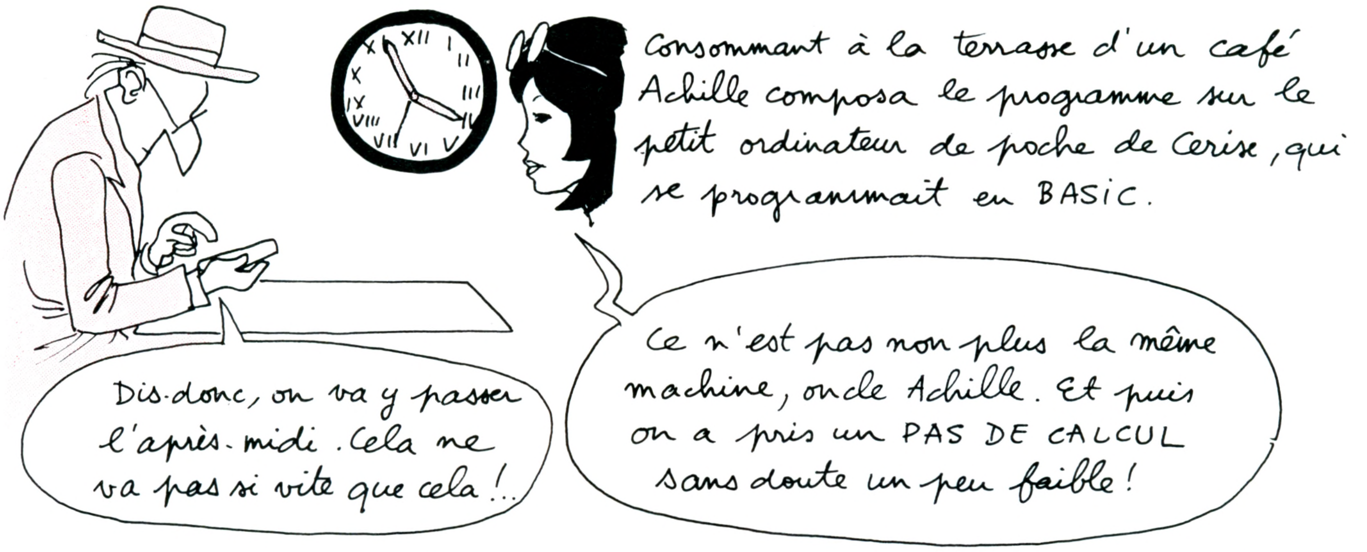
– Voilà. Plus qu'à résoudre le problème. Je serais curieux de voir comment tu vas t'y prendre.

– Exactement comme tu l'as défini. Il suffit de convertir en programme cette manière de procéder (récursive), cet ALGORITHME.

– Je suppose que tu vas prendre une valeur de départ pour RM.
– Tu remarqueras que RM a pour minimum αS. En effet, si $RM = \alpha S$ le temps de remboursement est.... infini, puisque tu ne fais que payer l'intérêt, mensuellement.
– Il faudrait commencer par $RM = \alpha S$ plus quelque chose. Mais quoi ? Le plus simple est de fixer une précision PR sur le calcul de RM, par exemple 100F. Tu peux donc prendre comme RM initial la valeur $\alpha S + 100$.
– On remboursera alors, mois après mois, en INCRÉMENTANT.
– Mais sans dépasser le nombre de mois NM envisagé. Au bout de ce laps de temps, si la dette n'est pas épongée, c'est que le RM choisi était trop faible. On l'accroitra donc de 100F, et ainsi de suite, jusqu'à ce que ça colle. Cela donnerait :

```
70 RM = S*A
80 RM = RM+100
90 M = 0 : S1 = S : REM INTERMÉDIAIRE CALCUL
100 S1 = S1*(1+A)-RM : M = M+1
110 IF M<>NM THEN 100
120 IF S1>0 THEN 80
130 PRINT "REMBOURSEMENT MENSUEL : "; RM
```

ligne 70 on calcule la valeur minimale de RM
ligne 80 on incremente de 100 (cent Francs)
Ligne 90 On commence un cycle. M est le nombre de mois, qui est mis à zéro. La dette sera S1, elle varie au cours du temps. On calcule S1 à la ligne 100. et on INCRÉMENTE M.
Ligne 110 si M est différent de NM, alors on passe un mois de plus (retour en 100). Sortie automatique de cette BOUCLE lorsque M = NM. On regarde alors (ligne 120) si la dette subsiste. Si oui, on accroit RM (retour en ligne 80), sinon on donne le résultat : ligne 130



– Oui, mais si on accroît le PAS, l'INCRÉMENT de RM, on va diminuer la précision du résultat !

– On va faire autrement. A la ligne 80, au lieu de 100 nous allons mettre par exemple 500. On va ainsi calculer le remboursement mensuel à 500 Francs près.

– Bon, ça va plus vite, et après ?

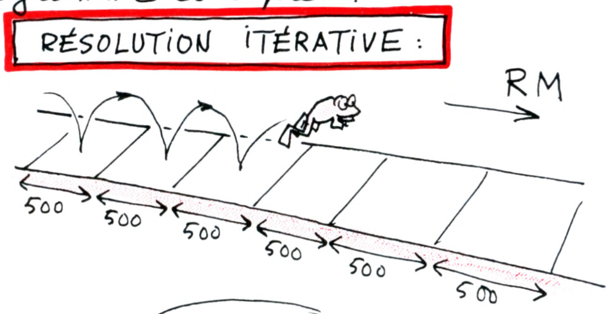
– Appelle R1 ce premier résultat. Tu sais alors que la solution exacte, au centime près, est entre R1 et R1 – 500. Tu prends donc R1 – 500 comme nouvelle valeur de départ pour le remboursement mensuel RM. Et tu réédites l'opération avec, cette fois, un incrément de cinquante. Tu obtiendras R2, valeur approchée à cinquante Francs près. Démarres un troisième calcul avec R2 – 50 comme valeur initiale, avec un incrément de cinq. Tu auras une nouvelle valeur du remboursement mensuel à cinq francs près.

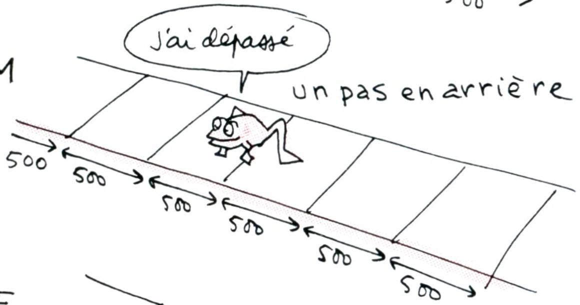
– Autrement dit, cette PROCÉDURE ITÉRATIVE m'amènera à un résultat précis en un temps cette fois acceptable ?

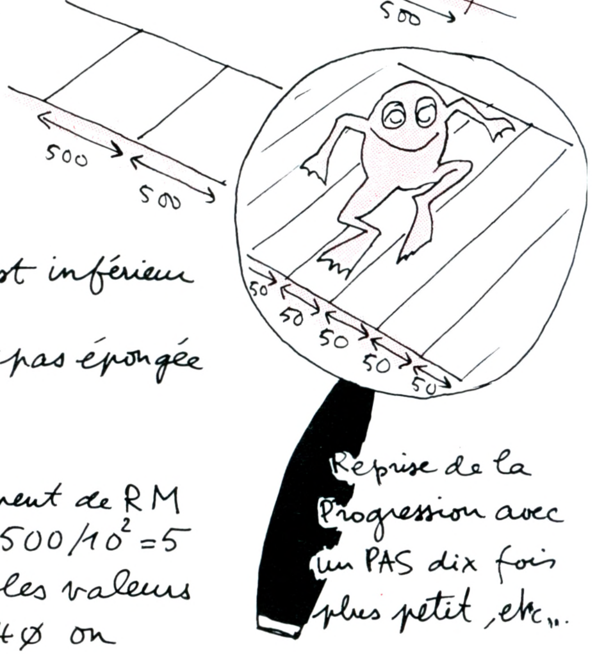
– Oui.

Cerise et Achille composèrent alors le programme ci-après :

```
10 [ Effaçage Ecran ]
20 INPUT "TAUX D'ESCOMPTE : "; TE
30 A = TE/1200 : PRINT
40 INPUT "SOMME EMPRUNTEE :"; S : PRINT
50 INPUT "NOMBRE D'ANNEES : "; NA
60 NM = NA*12 : REM NOMBRE DE MOIS
70 RM = S*A : REM VALEUR INITIALE
80 FOR I = 0 TO 2
90 IC = 500/(10^I) : REM INCREMENT RM
100 S1 = S : M = 0 : RM = RM + IC
110 S1 = S1*(1+A) - RM : M = M+1
120 IF M <> NM THEN 110
130 IF S1 > 0 THEN 100
140 RM = RM - IC : REM UN PAS EN ARRIERE
150 NEXT I : RM = RM + IC
160 PRINT "MENSUALITES DE : " RM; " F"
170 END
```







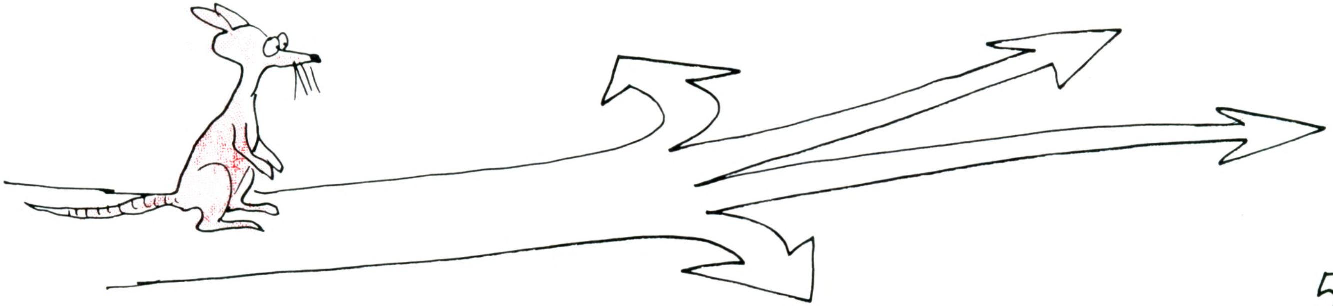
Il y a trois RETOURS EN BOUCLE imbriqués.
Ligne 120 : retour en 110 : si le nombre de mois est inférieur au nombre imposé NM.
Ligne 130 : Si au nombre de mois NM la dette n'est pas épongée on incrémente RM de IC en 100 et on recommence.
La boucle 80-150 a trois cycles (0, 1, 2). L'incrément de RM vaudra successivement $500/10^0 = 500$; $500/10^1 = 50$; $500/10^2 = 5$
La précision du calcul, le PAS, prendra donc les valeurs successives 500, 50, 5 (Francs). A la ligne 140 on fait un pas en arrière, pour repartir d'une valeur par défaut, avec un incrément dix fois plus faible

– Une autre question : quand on veut rembourser un emprunt en cours de route, on s'étonne souvent d'avoir "si peu remboursé".

– L'ordinateur peut te donner la réponse immédiatement. Il suffit de relancer le calcul et de stopper au bout d'un certain nombre de mois, ou d'années, et de voir ce qui reste à payer.

– Mais si je fais une étude sur un plan d'emprunt, et que je ne souhaite pas poser cette question à chaque fois ?

– Nous allons envisager maintenant des PROGRAMMES A OPTIONS.



NOTION DE MENU

On disait tout à l'heure que la programmation avait une parenté avec la cuisine. Ici, nous avons un menu à la carte. Si nous optons pour tel plat, cela va déclencher un certain nombre d'opérations en cuisine, qui aboutiront à la livraison du plat demandé.

– Imagine un restaurant qui ne va confectionner qu'un seul plat à la fois. Le travail accompli, le cuisinier reviendra te dire : « et ensuite ?... ». La suite pouvant éventuellement être : " merci, j'ai fini ".

– Ce restaurant devient un ordinateur ?!?

– Attends, regarde :

```
 10 [Effaçage Ecran]
 20 PRINT" VOULEZ-VOUS:":PRINT
 30 PRINT" 1-EMPRUNTER DE L'ARGENT":PRINT
 40 PRINT" 2-PLACER DE L'ARGENT":PRINT
 50 PRINT" 3-REMBOURSER EMPRUNT":PRINT
 90 PRINT" 9-SORTIR DU PROGRAMME":PRINT
100 INPUT"VOTRE CHOIX : ";R
110 IF R=9 THEN END
120 IF R=1 GOSUB 1000
130 IF R=2 GOSUB 2000
 :        :       :
200 GOTO 10
```

Sur l'écran je verrai apparaître :

VOULEZ-VOUS :
1 - EMPRUNTER DE L'ARGENT
2 - PLACER DE L'ARGENT
3 - REMBOURSER EMPRUNT
9 - SORTIR DU PROGRAMME
VOTRE CHOIX ■

SOUS PROGRAMMES

– Si ta réponse est 1 l'instruction GOSUB 1ØØØ va t'envoyer à la ligne 1ØØØ.

– Comme un GOTO 1ØØØ ?

– Non. Si ce SOUS-PROGRAMME 1ØØØ se termine par une instruction spéciale RETURN, le retour sera assuré à la ligne qui suit immédiatement le GOSUB, c'est à dire la ligne 13Ø.

Il est commode de mettre ses sous-programmes à des adresses simples : 1ØØØ, 2ØØØ, 3ØØØ, etc... et de "bloquer" à cet effet les lignes 1ØØØ à 1999, 2ØØØ à 2999, 3ØØØ à 3999, etc... On mettra systématiquement le RETURN aux lignes 1999, 2999, 3999, etc...

– Bon, il suffit que je reprenne le programme de la page 50 et que je le renumérote :

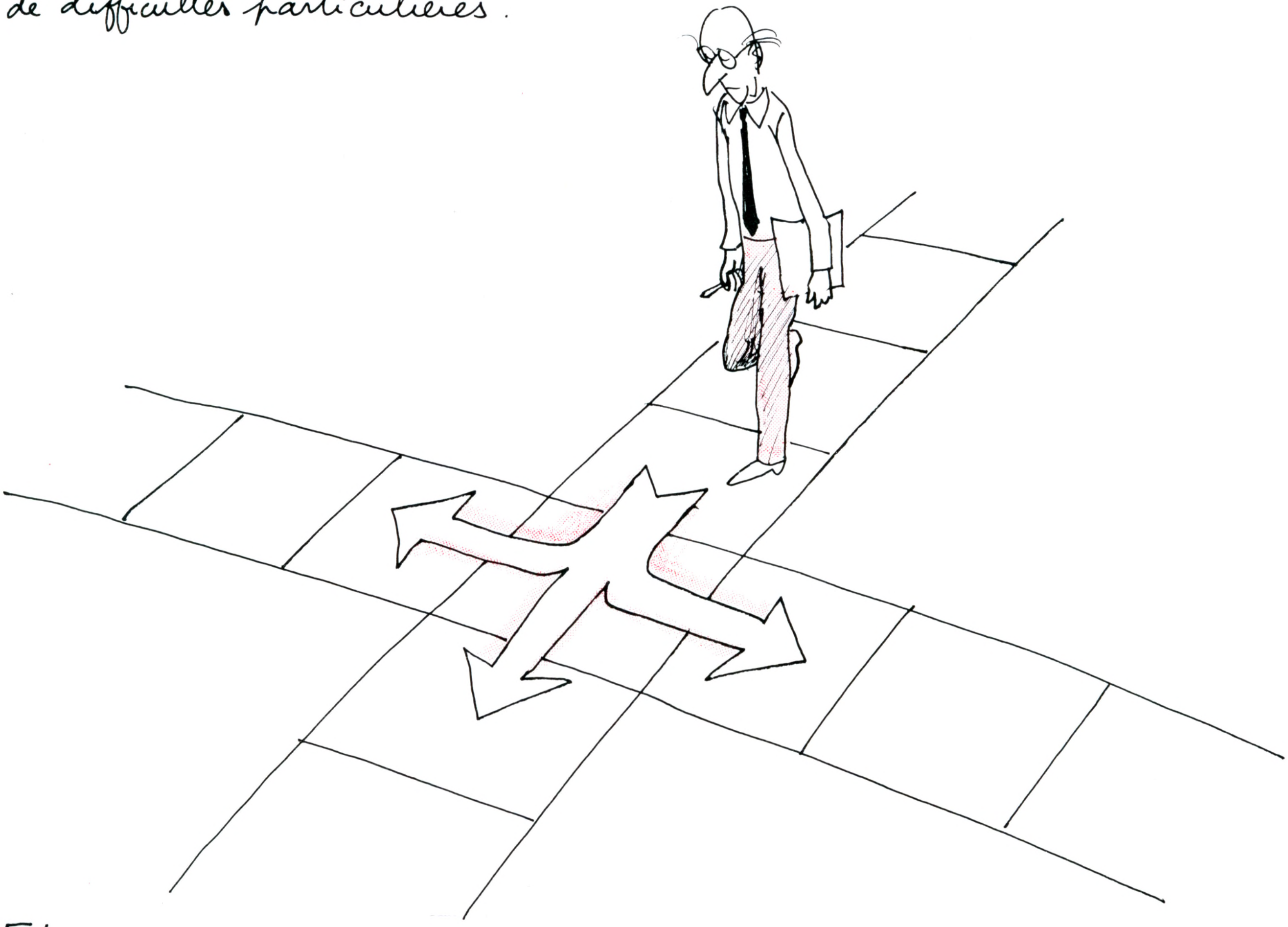
```
1Ø1Ø [Effaçage Ecran]
1Ø2Ø INPUT "TAUX D'ESCOMPTE";TE
1Ø3Ø A = TE/12ØØ : PRINT
     ETC... ETC...
1999 RETURN
```

– Pourquoi est-ce que les options sont numérotées 1, 2, 3, 9 ...?

– Ça te laisse la place pour en mettre d'autres.

– De même qu'il reste de la place entre 13Ø et 2ØØ, pour indiquer le sous programme correspondant. Cela n'a rien d'obligatoire, mais c'est commode d'associer l'option 4 à un sous-programme 4ØØØ-4999 l'option 5 à un sous programme 5ØØØ-5999, etc...

Moneyback compléta le programme présent dans l'ordinateur en introduisant effectivement les sous programmes 2 et 3, sans rencontrer de difficultés particulières.



— Je passe sur les détails. J'ai introduit dans l'ordinateur un numéro de sécurité sociale N

— Par où, mon ami ?

— Euh... mais comme cela, à l'aide des touches.

— A l'aide des touches ?...

— Maintenant il s'agit d'isoler le premier caractère, à gauche. Si c'est un 1 le sexe est masculin. Si c'est un 2, il est féminin. J'ai trouvé dans le manuel une instruction INT qui donne la partie entière d'un nombre.

— La partie entière ?...

— Alors j'écris :

```
1010 [Effaçage Ecran]
1020 INPUT "VOTRE NUMERO ?"; N : PRINT
1030 S = N / 10000 : S = INT (S)
1040 IF S = 1 THEN PRINT "SEXE MASCULIN" : GOTO 1070
1050 IF S = 2 THEN PRINT "SEXE FEMININ" : GOTO 1070
1060 GOTO 1020
1070 REM AMBIGUITE LEVEE
```

– Et pour la suite ?
– Eh bien je connais le premier chiffre du numéro, qui en comporte cinq. Je n'ai qu'à faire :

```
1080 A = N - S * 10000
```

Il ne me restera plus que les quatre nombres suivants. Si maintenant je divise par cent et que je prenne la partie entière :

```
1090 AN=INT(A/100)
```

J'aurai les deux chiffres qui indiquent l'année de naissance, et j'imprimerai :

```
1100 IF S=1 THEN PRINT"VOUS ETES NE EN 19";AN:PRINT
1110 IF S=2 THEN PRINT"VOUS ETES NEE EN 19";AN:PRINT
```

Puis je fais :

```
1120 B = A - AN * 100
1130 IF B = 1 THEN M$ = "JANVIER"
1140 IF B = 2 THEN M$ = "FEVRIER"
  :
1310 IF B=12 THEN M$="DECEMBRE"
1320 PRINT"AU MOIS DE ";M$

1999 RETURN
```

Cerise était arrivée sur ces entrefaites :

(*) Le lecteur complètera le programme en écrivant le sous programme de la 2e option et le "Menu"

TRAITEMENT DE TEXTE

– Mais on peut procéder autrement. Nous faisons ici une informatique CONVERSATIONNELLE où il y a échanges, dialogue, avec l'utilisateur à travers un système de questions - réponses. Mais l'ordinateur n'est qu'une machine. On ne peut pas dire qu'il comprenne ce qu'on lui dit. Il est donc essentiel d'éviter toute ambiguïté. Imagine une question à laquelle il faut répondre par OUI ou par NON. Écrivons par exemple :

```
510 INPUT "VOTRE RÉPONSE :"; R$
520 IF R$ = "OUI" THEN 110
530 IF R$ = "NON" THEN END
```

– Que se passera-t-il si l'utilisateur répond OK

Comme cette réponse n'est pas OUI l'ordinateur n'ira pas à la ligne 110.
Comme cette réponse R$ n'est pas NON, il ne s'arrêtera pas non plus. S'il existe en aval un sous programme quelconque, il l'exécutera comme un aveugle et finira par s'arrêter en affichant :

```
RETURN WITHOUT GOSUB
```

C'est à dire que ce sous programme aura été exécuté sans qu'il y ait eu APPEL de celui-ci par un ordre GOSUB, ce qui est illicite. On préférera écrire :

```
510 INPUT "VOTRE RÉPONSE :"; R$ : PRINT
520 IF R$ = "OUI" THEN 110
530 IF NOT(R$ = "NON") THEN 510
540 END
```

Imparable ! Si la réponse est OUI, branchement en ligne 110 (par exemple) Puis si la réponse N'EST PAS NON, reformulation de la question

– Mais revenons à cette histoire de numéro d'identification. Supposons qu'au lieu de le considérer comme un NOMBRE, on le considère comme une CHAÎNE DE CARACTÈRES. Et qu'on le SAISISSE à l'aide d'un INPUT N$

Primo, tu peux vérifier que l'utilisateur a bien donné cinq chiffres, ni plus, ni moins. Pour cela il suffit de compter le nombre de caractères de la chaîne N$, ce qui se fait immédiatement à l'aide de LEN (*)

```
10 INPUT "DONNEZ LES CINQ PREMIERS CHIFFRES
DE VOTRE NO D'IDENTIFICATION :"; N$ : PRINT
20 L = LEN (N$)
30 IF L<5 OR L>5 THEN 10
```

le sens de cette ligne 30 étant : si L est inférieur à 5 ou supérieur à 5

On aura aussi pu mettre IF L <> 5 (différent de) ou IF NOT (L=5) (si L n'est pas 5)

Ceci fait, l'instruction LEFT$ (left en anglais veut dire gauche) permet d'isoler un paquet de lettres en partant de la gauche, dans la chaîne de caractères N$.

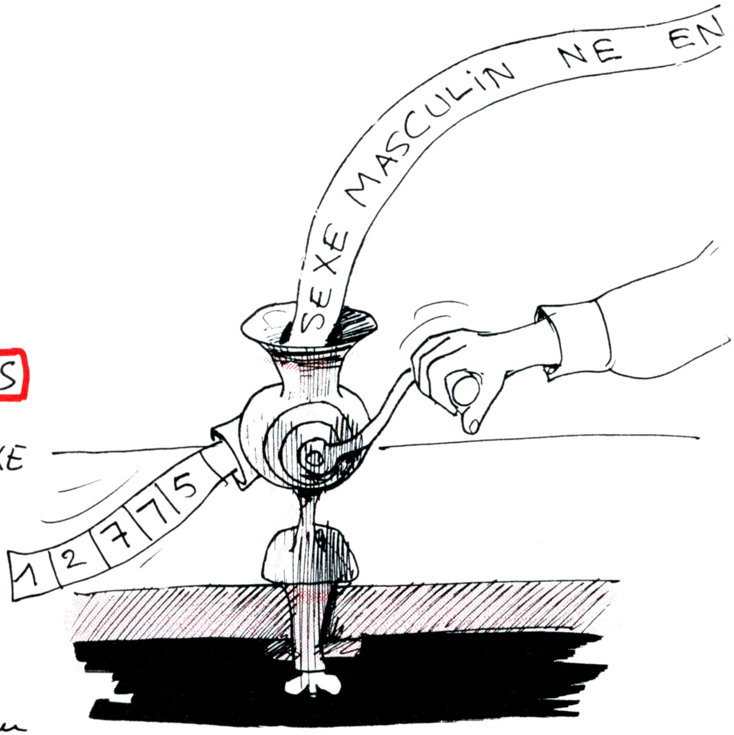
Il s'agit ici de prendre un caractère et on écrira :

```
40 S$ = LEFT$ (N$,1)
```

S$ sera une SOUS-CHAÎNE DE CARACTÈRES

Puis on enchaînera :

```
50 IF S$="1" THEN PRINT "SEXE
MASCULIN" : PRINT : GOTO 90
70 IF NOT (S$="2") THEN 10
80 PRINT "SEXE FEMININ"
90 etc... etc.
```

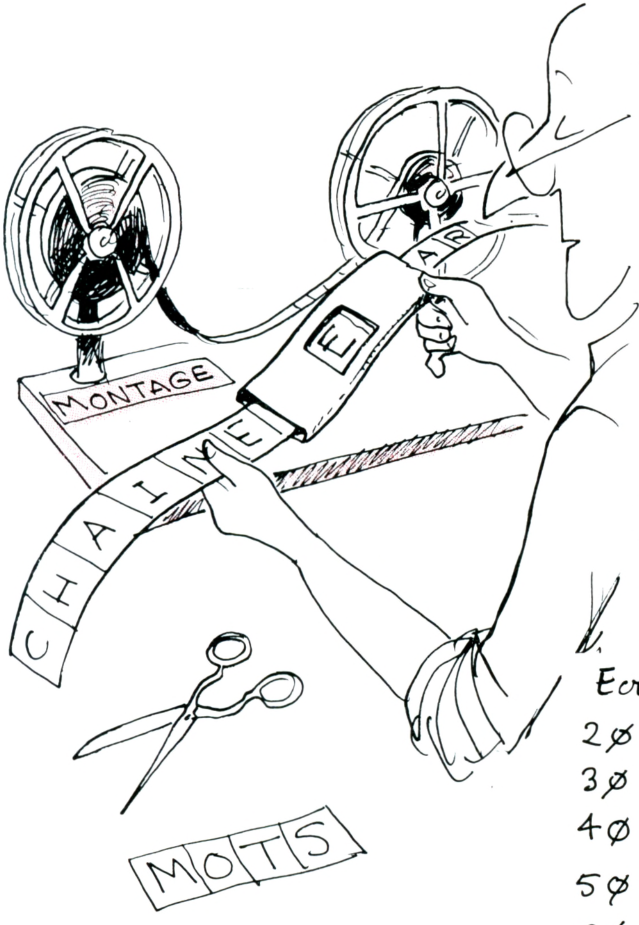


* de LENGHT, en anglais, longueur

Pour isoler le 2e et le 3e caractère, on utilisera une instruction MID$ (middle en anglais veut dire milieu).

```
90 AN$ = MID$(N$, 2, 2)
```

MID$(N$, P, Q) veut dire : isoler Q caractères à partir du P^ième (compris) en partant de la gauche, dans la chaine N$.
Ces groupes de lettres, ou de caractères, forment des SOUS-CHAINES.



– La communication avec l'ordinateur va devenir de plus en plus sophistiquée avec le temps. On cherche par exemple à mettre au point de (bonnes) machines à traduire. Il ne s'agit plus de traduire des mots, mais des phrases entières.
Le premier travail de l'ordinateur sera alors de ranger les mots d'une phrase, initialement separés par un "blanc" dans la chaine P$. On parcourt celle-ci, caractère par caractère, et dès qu'on rencontre un blanc, on s'arrête et on crée une sous chaine qui sera le mot isolé.

Ecrivons :

```
10 [Effaçage Ecran]
20 PRINT"VOTRE PHRASE :": PRINT
30 INPUT P$
40 P$ = P$ + " ":REM RAJOUTER BLANC
50 I = 0 :REM MISE A ZERO COMPTEUR
60 I = I + 1 : REM INCREMENTATION
```

```
70 C$ = MID$(P$,I,1)
```

Ceci isole le ième caractère de la chaine P$

```
80 IF C$<>" " THEN 60
```

Si C$ n'est pas un blanc alors aller au caractère suivant.

```
90 M$ = LEFT$(P$,I-1)
```

le ième caractère a été identifié comme blanc.

Donc les (I-1) premiers caractères de P$ forment un MOT.

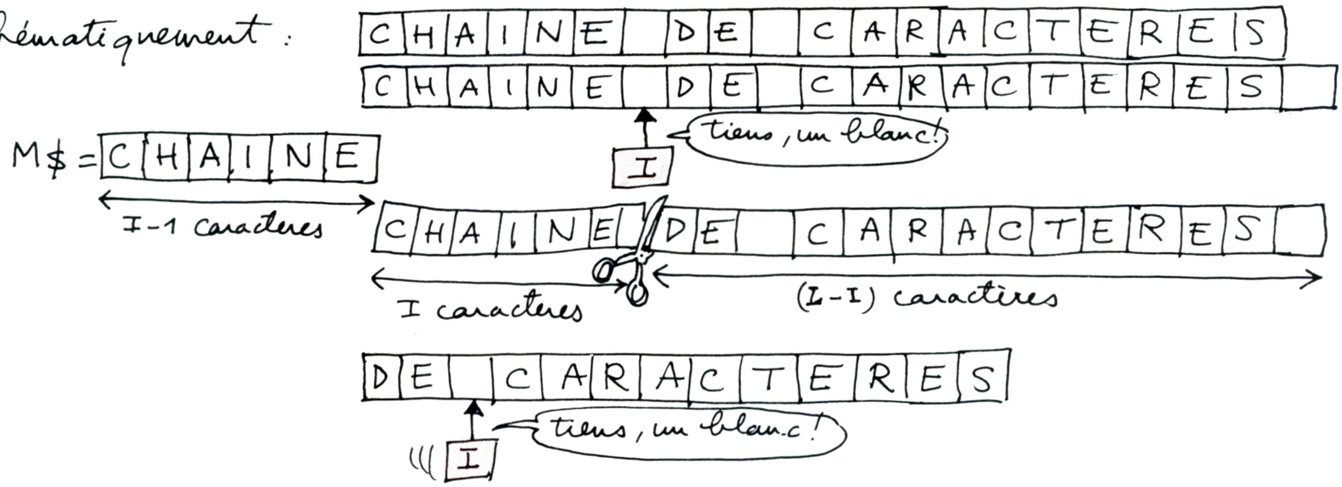
```
100 PRINT M$ : L=LEN(P$)
110 IF L=I THEN END
```

si le blanc identifié est le dernier caractère de la chaine, alors le travail est terminé.

```
120 P$ = RIGHT$(P$,L-I) :GOTO 50
```

Sinon on va amputer la chaine des I premiers caractères (le mot plus le blanc), garder la portion droite grâce à l'instruction RIGHT$ et revenir en 50 : nouvelle exploration de la chaine

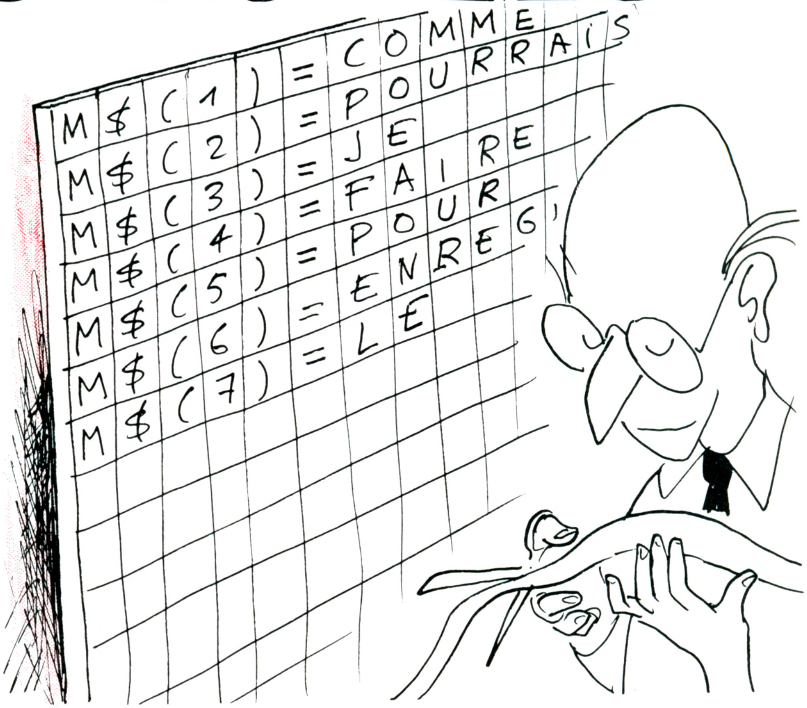
Schématiquement :



VARIABLES INDICÉES

— Comment pourrais-je faire, dit Achille, pour enregistrer le "premier" mot, puis le "second" etc... Il faudrait un chiffre, un INDICE, qui permettrait de se repérer. Par exemple l'indice J. On pourrait parler du $J^{ème}$ mot de la phrase.

— Pas de problème, c'est un langage que l'ordinateur comprend très bien. Tu peux manipuler **M$(J)**



Moneyback écrivit donc :

```
10 [Effaçage écran]
20 PRINT "VOTRE PHRASE :" : PRINT
30 INPUT P$   : PRINT
40 P$ = P$ + " " :REM RAJOUTER UN BLANC A DROITE
50 J = 0 :REM MISE A ZERO COMPTEUR MOTS
60 J = J + 1 : REM INCREMENTATION
70 I = 0 :REM MISE A ZERO COMPTEUR CARACTÈRES
80 I = I + 1 : REM INCREMENTATION
90 C$ = MID$(P$, I, 1) :REM ISOLER CARACTERE
100 IF C$ <> " " THEN 80
```

```
110 M$(J)=LEFT$(P$,I-1):REM ISOLER LE MOT
120 PRINT M$(J):PRINT
130 L=LEN(P$) :REM LONGUEUR CHAINE P$
140 IF L=I THEN END :REM FIN DE TACHE
150 P$=RIGHT$(P$,L-I):REM TRONQUER CHAINE
160 GOTO 60 :REM RECHERCHE DU MOT SUIVANT
```

Cette séquence a découpé la phrase en mots M$(J). On peut les réafficher en écrivant par exemple :

```
170 FOR K=1 TO J
180 PRINT M$(K):PRINT
190 NEXT K
```

– Mais alors, l'indice, c'est J ou c'est K ??

J contient, à la fin du découpage, le nombre de mots de la phrase. Si tu veux, tu pourrais écrire :

```
165 NM=J :REM NB DE MOTS
170 FOR J=1 TO NM
180 PRINT N$(J):PRINT
190 NEXT J
```

Comme cela tu peux réutiliser ta lettre J. Mais, pour l'ordinateur, ce qui compte, c'est la quantité numérique qui se trouve dans ces registres J ou K.

On a créé dans la mémoire une TABLE à un indice M$(1), M$(2), M$(3) etc... et c'est dans cette table que l'on puise, que cela soit avec un indice J ou K. Seul compte le NOMBRE contenu dans ces registres.

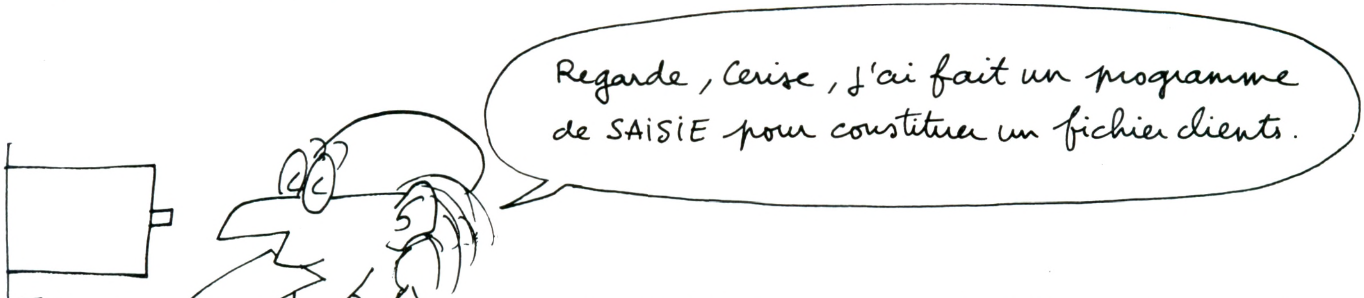
```
LES PANTOUFLES VOLAIENT DANS L'AZUR FAMELIQUE
LES
PANTOUFLES
VOLAIENT
DANS
L'AZUR
FAMELIQUE
```

voilà ce que cela donne

ÉLÉMENTS DE GESTION

Moneyback travailla une partie de la nuit, avec fièvre... Consultant fréquemment son manuel et pianotant nerveusement sur le clavier de la machine.

FICHIERS



```
1Ø1Ø REM LISTE CLIENTS + COMPTES
1Ø2Ø [Effaçage Ecran]
1Ø3Ø I = Ø
1Ø4Ø I = I + 1
1Ø5Ø INPUT "NOM CLIENT "; N$(I) : PRINT
1Ø6Ø PRINT "SON ADRESSE :" ; PRINT
1Ø7Ø INPUT "RUE, NUMERO "; R$(I)
1Ø8Ø INPUT "ARRONDISSEMENT : "; AR$(I)
1Ø9Ø INPUT "VILLE:" ; V$(I)
11ØØ INPUT " PAYS : "; P$(I) : PRINT
11Ø5 INPUT "COMPTE = "; CC(I) : PRINT
111Ø PRINT "IL S'AGIT BIEN DE : " : PRINT
112Ø PRINT N$(I)
113Ø PRINT R$(I)
114Ø PRINT V$(I); "  "; AR$(I)
115Ø PRINT P$(I)
116Ø PRINT "COMPTE ="; CC(I); " F" : PRINT
117Ø INPUT "CORRECT ?"; R$ : PRINT
118Ø IF R$ = "OUI" THEN 121Ø
119Ø IF NOT (R$ = "NON") THEN PRINT
"REPONDEZ PAR OUI OU NON" : GOTO 117Ø
12ØØ GOTO 1Ø5Ø
```

```
1210 PRINT:INPUT"UN AUTRE CLIENT?";R$:PRINT
1220 IF R$="OUI" THEN 1040
1230 IF NOT (R$="NON") THEN PRINT"RÉPONDEZ OUI OU NON":
 GOTO 1210
1240 NC=I :REM NOMBRE DE CLIENTS
```

— Oui, c'est pas mal. Tu crées des fichiers avec un indice de base, qui est le numéro du client. A partir de ce numéro tu pourrais aisément connaitre :

- N$(I) le nom du client
- R$(I) sa rue et son numéro
- AR$(I) son arrondissement
- V$(I) la ville où il réside
- P$(I) son pays
- CC(I) l'état de son compte (avoir ou crédit)

— Je devrais pouvoir inclure ce programme dans un programme principal, un MENU, regroupant les différentes tâches.

— Peux tu les décrire ?

— Pour commencer, après avoir CRÉÉ le fichier, il faut pouvoir le COMPLETER, puis le CONSULTER, METTRE A JOUR LES COMPTES DES CLIENTS (au fur et à mesure des facturations ou des règlements de factures). Il faut aussi pouvoir RELANCER LES IMPAYES, ANALYSER LES COMPTES...

— Et bien d'autres choses encore. Pour le moment, disons-nous que cela suffira. A titre d'exercice.

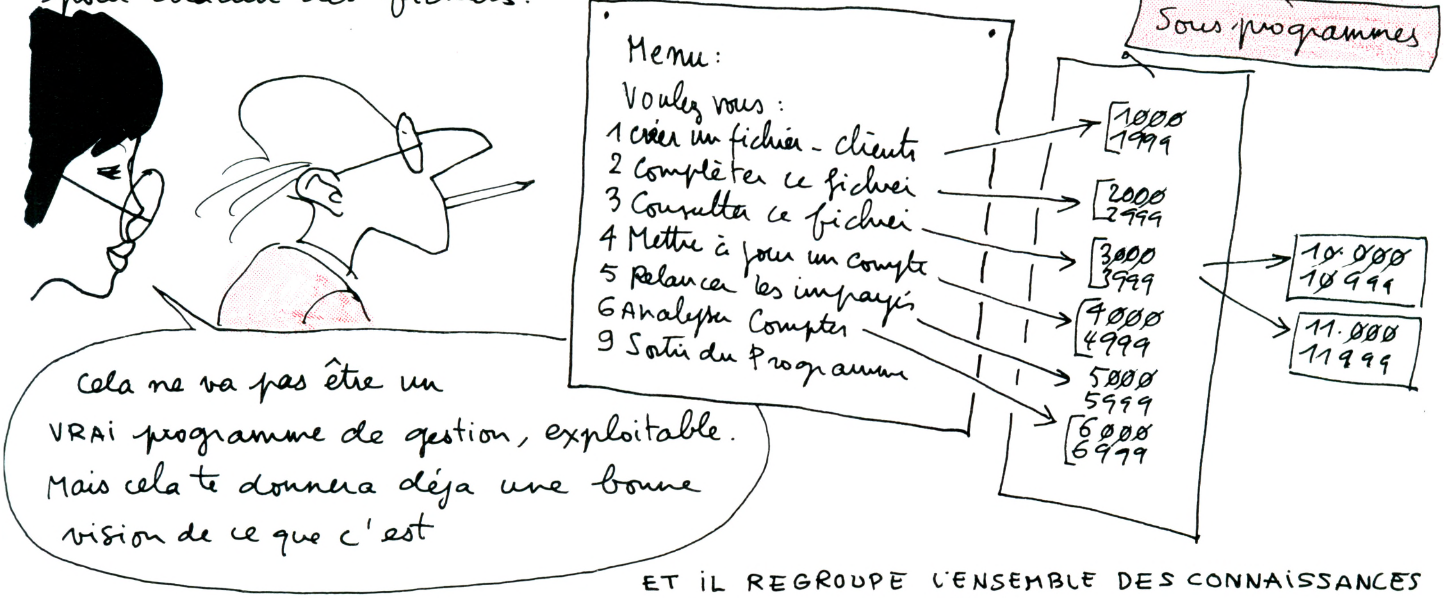
— Nous allons écrire le MENU ?

— Oui ...

— Combien as-tu de clients ?
— De gros clients, une cinquantaine...
— On va prendre cent. L'indice pourra aller jusqu'à cent. Il faut alors RESERVER DES PLACES EN MÉMOIRE.
— Que veux-tu dire ?
— Pour les variables indicées, tu disposes automatiquement de dix places correspondant aux valeurs d'indice : (Ø, 1, 2, 3, 4, 5, 6, 7, 8, 9).
Si la valeur de l'indice peut dépasser 9, comme c'est le cas ici, il faut le spécifier à l'aide d'ordres DIM (pour dimension)
Supposes que tu veuilles créer des tables, des fichiers N$(I), R$(I) AR$(I), V$(I), P$(I), CC(I).

On mettra : DIM N$(1ØØ), R$(1ØØ), AR$(1ØØ), V$(1ØØ), P$(1ØØ), CC(1ØØ)

Cette ligne réservera cent places (en fait cent dix car l'indice zéro compte) pour chacun des fichiers.

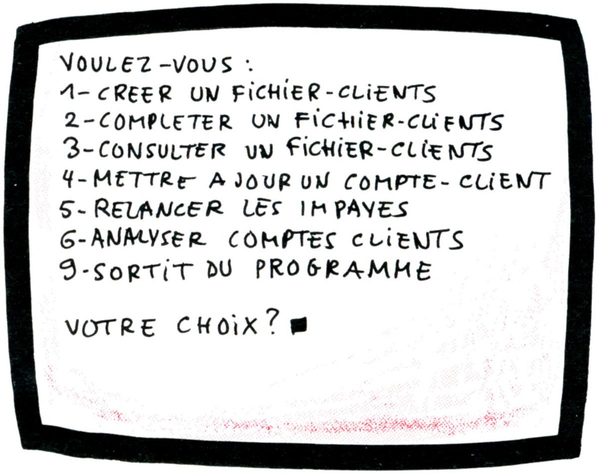


ET IL REGROUPE L'ENSEMBLE DES CONNAISSANCES ACQUISES JUSQU'ICI.

```
10 REM PROGRAMME DE GESTION DE COMPTES-CLIENTS
20 REM RESERVATION DE PLACES EN MEMOIRE :
30 DIM N$(100),R$(100),AR$(100),V$(100),P$(100),CC(100)
100 [Effaçage Ecran]

110 REM PROGRAMME-MENU :
120 PRINT"VOULEZ-VOUS :":PRINT
130 PRINT"1-CREER UN FICHIER-CLIENTS": PRINT
140 PRINT"2-COMPLETER UN FICHIER-CLIENTS":PRINT
150 PRINT"3-CONSULTER UN FICHIER-CLIENTS":PRINT
160 PRINT"4-METTRE A JOUR UN COMPTE-CLIENT":PRINT
170 PRINT"5-RELANCER LES IMPAYES" : PRINT
180 PRINT"6-ANALYSER COMPTES CLIENTS":PRINT
190 PRINT"9-SORTIR DU PROGRAMME":PRINT
200 INPUT"VOTRE CHOIX :"; C : PRINT
210 IF C = 9 THEN END
220 IF C = 1 THEN GOSUB 1000
230 IF C = 2 THEN GOSUB 2000
240 IF C = 3 THEN GOSUB 3000
250 IF C = 4 THEN GOSUB 4000
260 IF C = 5 THEN GOSUB 5000
270 IF C = 6 THEN GOSUB 6000
300 GOTO 100
310 REM RETOUR AU MENU
```

Sur l'écran !



– Bon, j'ai déjà écrit le sous-programme 1000.
– Mais, que fais-tu si tu te trompes ? si tu fais une erreur de frappe ?
– Ce serait très ennuyeux...
– On va mettre un petit circuit de vérification avant stockage définitif, qui limitera considérablement les erreurs.

```
1000 REM CONSTITUTION FICHIER-CLIENTS
1010 [Effaçage Ecran]
1020 REM MISE A ZERO DU NUMERO CLIENT :
1030 I = 0
1040 I = I + 1 : REM INCREMENTATION
1045 IF I>100 THEN PRINT "DEPASSEMENT DE CAPACITE-MEMOIRE" :
GOSUB 20000 : GOTO 1999
```

– Qu'est-ce qu'il y aura à cette adresse 20.000 ?
– Un petit sous-programme qui comportera une boucle de temporisation. Puis on renvoit en 1999 RETURN. Inutile de poursuivre la tâche puisqu'elle est considérée comme impossible. On saute donc en fin de sous-programme

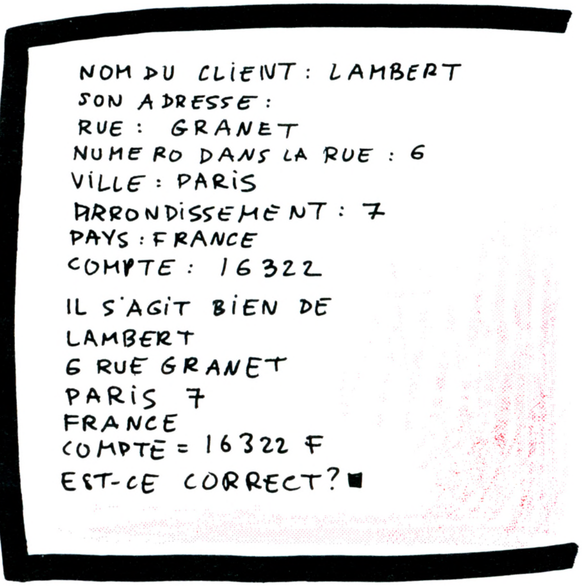
```
1050 INPUT "NOM DU CLIENT :"; N$(I) : PRINT
1060 PRINT "SON ADRESSE : " : PRINT
1070 INPUT " RUE : "; R$ : PRINT : REM DONNEE PROVISOIRE
1080 INPUT "NUMERO DANS LA RUE : "; N$ : PRINT
1087 R$(I) = N$ + " RUE " + R$
```

– Là nous faisons une CONCATENATION.
CATENA veut dire chaîne en latin. Littéralement "mettre en chaîne". C'est l'opération inverse des troncatures MID$ ou LEFT$ ou RIGHT$

L'ORDINATEUR NE CONFOND PAS R$ et R$(I), CONSIDÉRÉS COMME DISTINCTS

Tu as un certain nombre de chaines de caractères (ici N$. "RUE" et R$, et l'aide du + tu formes une chaine par association de ces trois.

```
1090 INPUT"VILLE:";V$(I):PRINT :INPUT"ARRONDISSEMENT:";AR$(I):PRINT
1100 INPUT"PAYS:";P$(I):PRINT
1105 INPUT"COMPTE:";CC(I):PRINT
1110 PRINT"IL S'AGIT BIEN DE:":PRINT
1120 PRINT N$(I)
1130 PRINT R$(I)
1140 PRINT V$(I);"  ";AR$(I)
1150 PRINT P$(I)
1160 PRINT "COMPTE=";CC(I);" F" :PRINT
1170 INPUT"EST-CE CORRECT ?";R$ :PRINT
1180 IF R$="OUI"THEN 1210
1190 IF NOT(R$="NON")THEN PRINT "REPONDEZ PAR OUI OU PAR
NON" : GOTO 1170
1200 GOTO 1050 :REM SI ERREUR
1210 INPUT "UN AUTRE CLIENT?";R$ : PRINT
1220 IF R$="OUI"THEN 1040
1230 IF NOT(R$="NON")THEN PRINT
"REPONDEZ OUI OU NON": GOTO 1210
1240 NC = I : REM NB DE CLIENTS
1999 RETURN
```



– Ce petit système permet de se rattrapper en cas d'erreur. On repart alors en ligne 1050 c'est à dire qu'on recommence

à entrer les données, mais avec le même indice I. Il s'agit toujours du même client. Par contre à la ligne 1220, si la réponse est OUI on renvoit à la ligne 1040 où on incrémente I, c'est à dire qu'on passe au client suivant.

Ce programme de création de fichier-clients ne va te servir qu'une fois. Ultérieurement tu te contenteras de le compléter, selon l'option 2. On pourrait s'arranger même pour que l'option 1, après usage, disparaisse du menu.

— Parce que si je fais l'erreur de m'orienter vers cette option le premier ordre met à zéro le numero-client. Tout est fichu en l'air...

— Tu vois donc qu'il y a un gros problème de fiabilité. Il faut avoir pensé à tous les cas possibles, à toutes les erreurs.

le sous programme 2000 est dérivé du précédent.

```
2000REM  COMPLETER FICHIER
2010 [Effaçage Ecran]
2020 PRINT"CE FICHIER CONTIENT :";NC;" CLIENTS" :PRINT
2030 NC=NC+1 :REM INCREMENTER NC
2040 PRINT"NOUVEAU CLIENT:" :PRINT
2050 INPUT"NOM : ";N$(NC) : PRINT
2060 PRINT"SON ADRESSE :"; PRINT
2070 INPUT"RUE:";R$(NC) : PRINT
2075 INPUT"NUMERO DANS LA RUE";N$:PRINT
2080 R$(NC)=N$+" RUE "+R$
2085 INPUT"VILLE:";V$(NC) : PRINT
2090 INPUT"ARRONDISSEMENT:";AR$ :PRINT
2100 INPUT"PAYS"; P$(NC): PRINT
2105 INPUT"COMPTE =";CC(NC) :PRINT
2110 PRINT "IL S'AGIT BIEN DE:":PRINT
2120 PRINT N$(NC)
2130 PRINT R$(NC)
2140 PRINT V$(NC);" ";AR$(NC)
2150 PRINT P$(NC)
2160 PRINT"COMPTE =";CC(NC);" F" : PRINT
2170 INPUT"CORRECT ?";R$ :PRINT
2180 IF R$="OUI"THEN 2210
2190 IF NOT(R$="NON")THEN PRINT"REPONDEZ OUI OU NON" :
PRINT: GOTO 2170
2200 GOTO 2050
2210 INPUT"UN AUTRE CLIENT?";R$ :PRINT
```

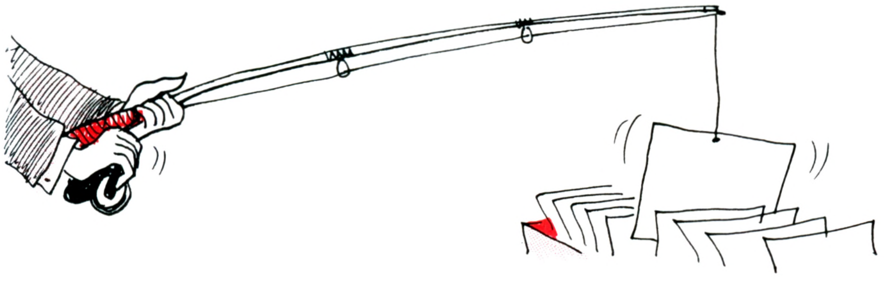
```
2220  IF R$="OUI" THEN 2040
2230  IF NOT(R$="NON") THEN PRINT"REPONDEZ OUI OU NON":PRINT:GOTO2210
2999 RETURN
```

— Passons à la consultation du fichier :

```
3000 REM  CONSULTATION FICHIER-CLIENTS
3010 [Effaçage Ecran]
3020 PRINT" VOULEZ-VOUS :" : PRINT
3030 PRINT"1- EXAMINER UN COMPTE CONNAISSANT LE" : PRINT :
PRINT"NOM DU CLIENT" : PRINT : PRINT
3040 PRINT"2- IDEM EN CONNAISSANT SON NUMERO" : PRINT
3050 INPUT"VOTRE CHOIX ?"; C
3060 IF C = 1 THEN GOSUB 10000
3070 IF C = 2 THEN GOSUB 11000
3999 RETURN
```

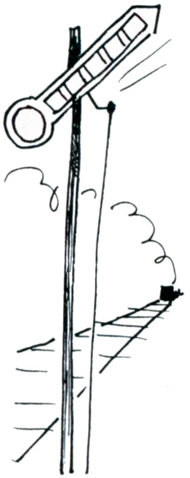
— Il y a un deuxième MENU !?!

— Oui, c'est un SOUS-MENU. Et il envoie à des SOUS-OPTIONS et à deux autres sous-programmes (*) que nous préciserons plus loin. Ils vont correspondre à l'identification du client dans le fichier.



(*) On devrait dire à des sous-sous-programmes...

```
4000 REM  MISE A JOUR COMPTE-CLIENT
4010 [Effaçage Ecran]
4020 PRINT "1-VOUS CONNAISSEZ LE NOM DU CLIENT" : PRINT
4030 PRINT "2-VOUS CONNAISSEZ SON NUMERO" : PRINT
4040 FL = 0 : INPUT "VOTRE CHOIX ?"; C : PRINT
4050 IF C = 1 THEN GOSUB 10000
4055 IF C = 2 THEN GOSUB 11000
4060 IF FL = 0 THEN PRINT "ERREUR" : GOSUB 20000 : GOTO 4999
4070 [Effaçage Ecran]
4080 PRINT "IL S'AGIT BIEN DE :"; N$(I) : PRINT
4090 PRINT "COMPTE ="; CC(I); " F" : PRINT
4100 PRINT "INDIQUEZ L'OPERATION, POUR UN REGLEMENT" : PRINT
4110 PRINT "FAITES PRECEDER D'UN +" : PRINT
4120 PRINT "POUR UN CREDIT, D'UN -" : PRINT
4130 INPUT "MONTANT OPERATION :"; OP : PRINT
4140 CC(I) = CC(I) + OP
4150 PRINT "NOUVEL ETAT DE COMPTE :"; CC(I); " F" : GOSUB 20000
4999 RETURN
```



— Que vient faire ce FL = 0, puis ce texte, ligne 4060 ?

— C'est un FLAG. On commence par le mettre à zéro. On le retrouvera dans les sous programmes 10 000 et 11 000. Si le client est identifié, par son nom ou son numéro, alors le Flag (Registre mémoire FL) sera mis égal à un. C'est un souvenir d'une opération effectuée. N'importe quel registre peut servir de Flag

— Et on peut faire le test sur ce qu'on veut.

— Eventuellement sur une gamme de valeurs qui mémorisent un choix parmi plusieurs options.

— Et pour relancer mes impayés ?

```
5000 REM  RELANCE  IMPAYES
5010 [ Effaçage Ecran ]
5020 PRINT "SONT DEBITEURS :" : PRINT
5025 FOR I = 1 TO NC : REM EXPLORATION FICHIER
5030 IF CC(I)<0 THEN PRINT N$(I),P$(I),CC(I) : PRINT
```

— c'est un système (la virgule) qui permet d'afficher plusieurs renseignements sur une même ligne. Ici le nom, le pays, la somme due.

```
5040 NEXT I
5045 PRINT
5050 INPUT"APPUYEZ SUR LA TOUCHE (FIN DE MESSAGE)"; B$
5999 RETURN
```

— Cette ligne 5045 met l'ordinateur en attente d'une donnée, d'une chaîne de caractères, qu'il mettra dans un registre B$. Si on appuie directement sur la touche [Fin de message] une chaîne "VIDE" sera stockée.

— Pourquoi faire ???

— Rien... c'est une des multiples astuces pour stopper l'ordinateur (pour noter ce qu'il fournit). Il repartira sur une simple pression de la touche [Fin de message]

— Commode, en effet.

```
6000 REM ANALYSE (SOMMAIRE) DES COMPTES-CLIENTS
6010 [Effaçage Ecran] : A = 0 : P = 0
6020 PRINT "NOMBRE DE CLIENTS :"; NC : PRINT
6030 FOR I = 1 TO NC : REM EXPLORATION FICHIER
6040 IF CC(I) > 0 THEN A = A + CC(I)
```

... si le compte CC(I) du I ème client est en positif, on le somme sur l'ACTIF A

```
6050 IF CC(I) < 0 THEN P = P + CC(I)
```

... en négatif, sommation sur le PASSIF

```
6060 NEXT I
6070 PRINT : PRINT "ACTIF ="; A ; " F" : PRINT
6080 PRINT : PRINT "PASSIF ="; P ; " F"
6090 GOSUB 20000 : REM TEMPORISATION
6999 RETURN
```

– Il nous reste à écrire les sous programmes 10000, 11000 et 20000

```
10000 REM EXAMEN DE COMPTE, NOMINATIF
10010 [Effaçage Ecran]
10020 INPUT "NOM DU CLIENT :"; N$ : PRINT
10030 I = 0
10040 I = I + 1 : REM INCREMENTATION
10050 IF N$ = N$(I) THEN FL = 1 : PRINT "COMPTE ="; CC(I)
; " F" : GOSUB 20000 : GOTO 10999
10060 I=NC THEN PRINT "NOM HORS FICHIER" : GOSUB 20000 : GOTO 10999
1070 GOTO 10040
10999 RETURN
```

Cerise commenta :

– ligne 10050 si le client est reconnu, le Flag FL est mis à 1, et on indique son état de compte. Sortie de boucle. le registre I mémorise le numéro du client.

```
11000 REM CONSULTATION COMPTE PAR NUMERO
11010 [Effaçage Ecran]
11020 INPUT"NUMERO DU CLIENT?";I:PRINT
11030 IF  >NC THEN PRINT"ERREUR,NUMERO TROP GRAND":
 GOSUB 20000 : GOTO 11999
11040 PRINT"NOM DU CLIENT:";N$(I) :PRINT
11045 PRINT"COMPTE:";CC(I); " F":GOSUB 20000 :FL=1
11999 RETURN
```

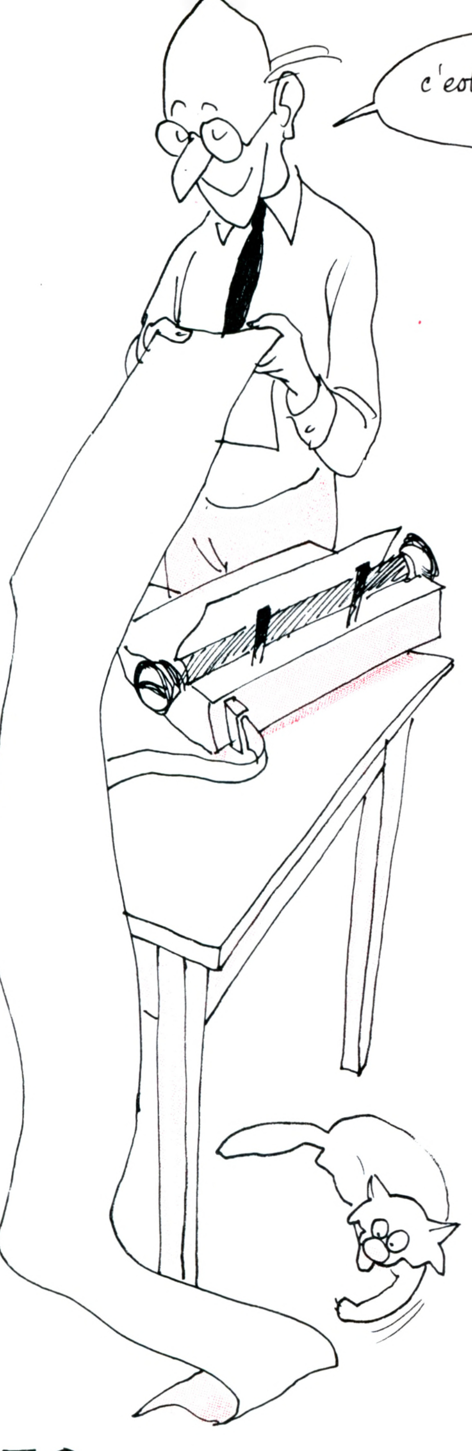
Encore une fois le Flag indique si le client a été trouvé et le registre I contient son numéro, correct ou non.

– Reste la boucle de Temporisation :

```
20000 REM TEMPORISATION
20005 FOR T= 1 TO 2000 : NEXT T
20999 RETURN
```

– Tu mets toute la boucle sur une seule ligne ?

– Aucun problème...



— Un petit début, oncle Achille. Mais tout cela t'a quand même appris les grandes lignes de la programmation. Il faudrait encore beaucoup de choses pour compléter ce programme. Par exemple introduire quelquechose pour transférer les fichiers sur disques.

— Mais... je n'ai qu'à faire SAVE ?

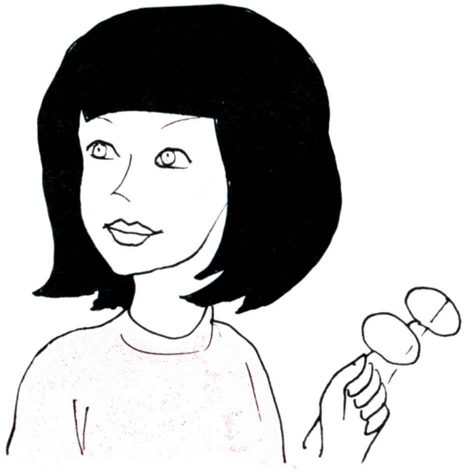
— Ce faisant, tu stockes le PROGRAMME, mais pas les contenus de mémoires.

— Ah... bon !

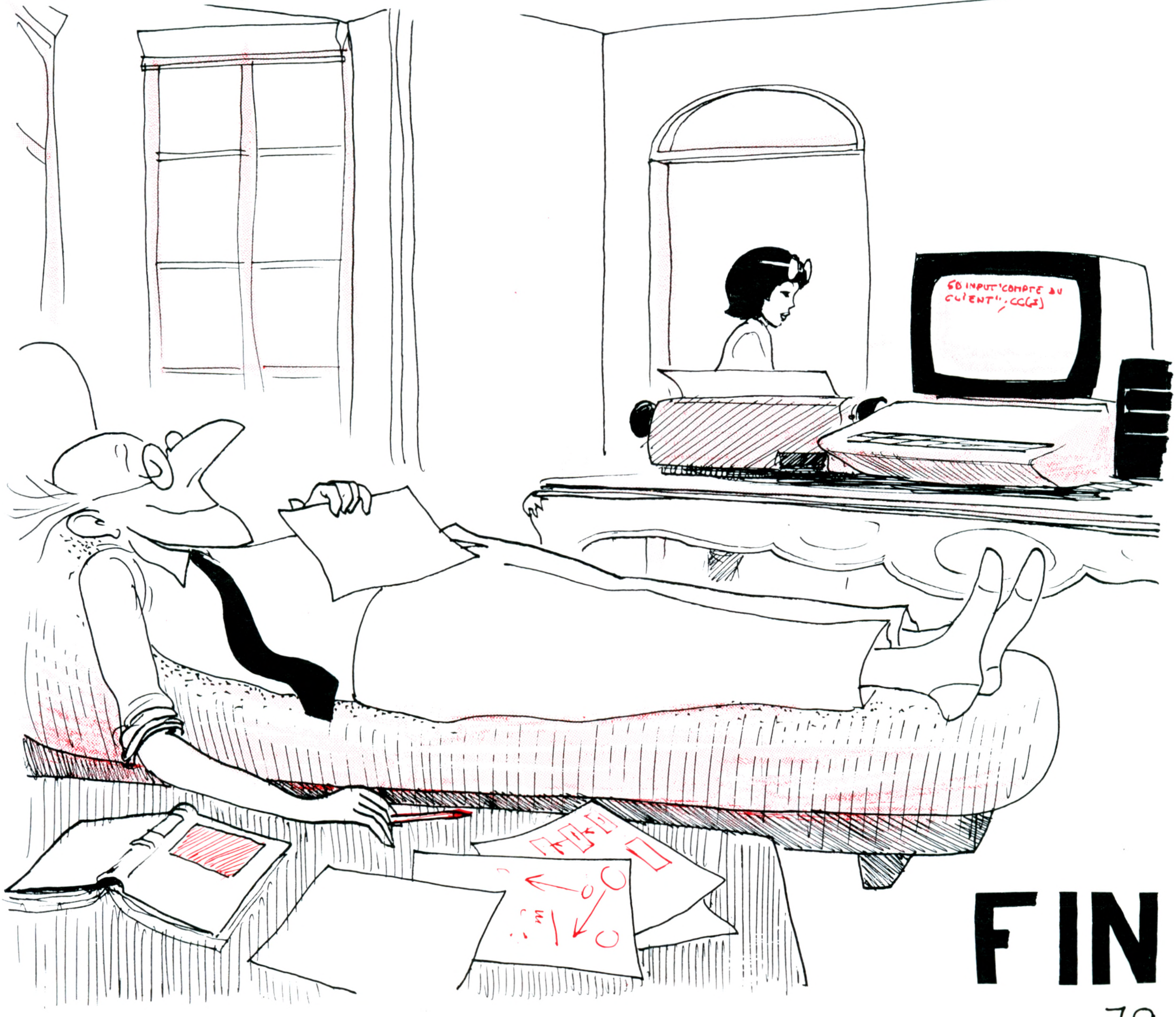
— Et rappelle-toi que lorsque tu fais RUN, la première chose qui se passe, c'est la remise à zéro de toutes les mémoires.

— Alors j'ai travaillé pour rien!

— Non, il manque certaines choses à ce programme. Mais ceci est une autre histoire...



La journée s'annonçait belle. Marie-Thérèse alla préparer un peu de café.
Cerise éteignit l'ordinateur et partit sur la pointe des pieds.



FIN

INSTRUCTIONS PRINCIPALES DU BASIC

On a indiqué, dans la colonne de gauche, les pages où ces instructions ont été introduites, au fur et à mesure, et décrites.

13	C=3 ou A=-8	Indique que les valeurs 3 et moins 8 sont inscrites dans les registres C et A
14	PRINT B / ?B	lecture du contenu d'un registre numérique B.
14	ENTER / RETURN	Instruction de fin de message.
14	CLEAR	met à zéro tous les contenus des registres, numériques ou littéraux.
15	SYNTAX ERROR	Message d'erreur le plus courant
16	N$="CERISE"	création d'un registre N$ contenant la chaine de caractères CERISE
22	END	Instruction d'arrêt
23	HOME	Pour effacer l'écran
23	1 PRINT "CLAVIER EST UN ANE"	ligne programme BASIC
24	NEW	Efface le programme
27	REM Article CP90	Permet d'inscrire des notes dans un programme.
27	LIST	déclenche l'affichage de tout programme présent en mémoire.
27	LIST 3Ø	Affiche seulement la ligne 3Ø.
27	LIST 5, 8Ø	Affiche les lignes 5 à 8Ø, incluses.
28	DEL 5Ø,11Ø	Efface les lignes 5Ø à 11Ø, incluses
3Ø	INPUT N	Exécution du programme interrompue pour attente d'une donnée numérique.
3Ø	INPUT N$	Même chose avec attente d'une chaine de caractères.
3Ø	INPUT "DATE LETTRE CLIENT:"; D$	Idem avec questionnement.

31	14Ø PRINT "QUANT À L'ARTICLE "; AR$; " NOUS NE " : PRINT	
	Façon d'afficher plusieurs chaines sur l'écran, bout à bout.	
34	SAVE · LOAD · CSAVE · CLOAD	recopie ou charge (sur, à partir de) la mémoire externe.
37	N=N+1	Incrémentation dans le registre N, l'incrément étant de 1.
38	5Ø GOTO 3Ø	branchement inconditionnel à la ligne 3Ø
40	5Ø IF N=15Ø THEN 7Ø	Branchement conditionnel (avec Test) vers ligne 7Ø
41	2Ø FOR N=2 TO NM	Début de boucle FOR NEXT (de N=2 à la valeur NM).
41	4Ø NEXT N	Renvoit à la ligne d'initiation de la boucle.
53	GOSUB 1ØØØ	Envoit vers le sous programme débutant à la ligne 1ØØØ
54	1999 RETURN	indique une fin de sous programme.
56	AN=INT(A)	AN= partie entière de A.
58	L=LEN(N$)	L est le nombre de caractères de la chaine N$
59	AN$=MID$(N$,2,2)	création de la sous-chaine AN$ à partir de N$
6Ø	M$=LEFT$(P$,5)	Idem. les 5 caractères, en partant de la gauche de P$.
6Ø	P$=RIGHT$(P$,4)	Idem. les quatre caractères, en partant de la droite de P$.
61	M$(J) · N(I)	Variables indicées.
65	IF NOT(R$="NON") THEN 11Ø	Test avec branchement sur réponse négative.
67	DIM N$(4ØØ), CC(4ØØ)	Réservations de place mémoire pour fichiers.

Imprimerie VAUDREY - LYON
N° d'édition 28637
Dépôt légal Mars 1984

EDITIONS CEDIC
DIFFUSION FERNAND NATHAN
32, Bd Saint-Germain 75005 Paris

F. 42/84/3

ISBN 2-7124-0608-7

L'INFORMATIQUE (premier contact)

Jean-Pierre Petit

CEDIC/FERNAND NATHAN